1997

Some Applications of Generalized FFTs

Daniel N. Rockmore
Dartmouth College

Some applications of generalized FFTs

Daniel N. Rockmore

Technical Report PCS-TR96-282

Revised 7/96

DIMACS Series in Discrete Mathematics
and Theoretical Computer Science
Volume 00, 19xx

Some applications of generalized FFTs

Daniel N. Rockmore

ABSTRACT. Generalized FFTs are efficient algorithms for computing a Fourier transform of a function defined on finite group, or a bandlimited function defined on a compact group. The development of such algorithms has been accompanied and motivated by a growing number of both potential and realized applications. This paper will attempt to survey some of these applications. Appendices include some more detailed examples.

1. A brief history

The now "classical" Fast Fourier Transform (FFT) has a long and interesting history. Originally discovered by Gauss, and later made famous after being rediscovered by Cooley and Tukey [21], it may be viewed as an algorithm which efficiently computes the discrete Fourier transform or DFT. In between Gauss and Cooley-Tukey others developed special cases of the algorithm, usually motivated by the need to make efficient data analysis of one sort or another. To cite but a few examples, Gauss was interested in efficiently interpolating the orbits of asteroids [43]; Danielson and Lanczos were concerned with x-ray diffraction [23]; Yates [103] and Good [47] needed the algorithm for statistics; Cooley and Tukey were interested in efficient time series analysis and digital signal processing [21]. For thorough historical overviews see [19, 20, 50].

Recently, there has developed a growing literature related to the construction of algorithms which generalize the FFT from the point of view of the theory of group representations (see e.g., [5, 17, 18, 29, 82]). These sorts of generalizations are "natural" as mathematical constructs, but in point of fact, they too have been motivated by applications. For example, the seemingly earliest construction of "nonabelian" FFTs (due to Willsky) was motivated by the search for new efficient filters [102]. Later constructions have been motivated by applications such as efficient data analysis (cf. [26]) and circuit design (cf. [6]), just to name a few examples. The purpose of this paper is to survey some of the applications of generalized FFTs and thereby (hopefully!) motivate further work in this direction.

1991 *Mathematics Subject Classification.* Primary 20C40, 68-02; Secondary 62-07, 62K10, 65T10, 68Q40, 42C10.

NSF DMS 9404275, 9304580, and Presidential Faculty Fellowship, AFOSR DOD F4960-93-1-0567.

1052-1798/00 $1.00 + $.25 per page

One early version of the FFT is due to the statistician Yates. He was interested in the efficient analysis of data from factorial designs. Section 2 reviews this algorithm and then explains in some detail its generalization in the form of efficient computation of spectral analysis for data on a finite group or its quotient. This is illustrated by a brief discussion of one of the more successful applications to date of these methods, that of the analysis of ranked and partially ranked data, for which Appendix A works out an example. Applications to the analysis of data from designs is also discussed. In this case, under some natural hypotheses, a decomposition is obtained which refines the usual ANOVA sum of squares decomposition.

Section 3 begins by recalling the Cooley-Tukey FFT and its application to time series analysis and convolution analysis via smoothing and matched filtering. The appropriate generalization to other groups and the use of nonabelian filters is then discussed. Of particular interest are the applications of the recently discovered fast convolution algorithm for bandlimited functions on the 2-sphere (cf. [**31, 55, 57**]), which in turn gives rise to the possibility of an efficient matched filter for the sphere. The discovery of an FFT for the 2-sphere by Driscoll and Healy was the first instance of a nonabelian FFT in the continuous setting. This algorithm and its successors have many other possible applications as well [**55, 57, 53**]. These are collected together briefly in Appendix B.

FFTs for finite groups make use of "symmetry-adapted representations". These representations are useful for FFT-like applications and some of these are discussed briefly in Section 4. Concluding remarks are in Section 5.

As the focus of this paper is on *applications* of generalized FFTs, details of the algorithms will not be discussed here. The interested reader should see [**81**] for an overview of some of these techniques as well as a fairly comprehensive bibliography. The recent book [**18**] is also a good place to look for both algorithms and applications (esp. Chapters 10 and 11).

Acknowledgement. Thanks to Persi Diaconis, Mark Goresky, Dennis Healy and Peter Kostelec for reading various pieces of preliminary versions. Thanks also to Rosemary Bailey, James Cooley, Geoff Davis, Peter Kim and Tim Olson for helpful conversations, Doug Warner for help with figures and to Georgia Thompson for sharing her data.

2. Spectral analysis

2.1. The basic idea. **Spectral analysis** is a non-model based approach to data analysis, formulated in a general group theoretic setting by Diaconis (cf. [**26**], esp. Chapter 8 as well as the extensive bibliography), which builds upon work of Hannan [**49**] and James [**61**], and extends the classical spectral analysis of time series. In brief, the idea is that often data has natural symmetries, encapsulated in the existence of a symmetry group for the domain of the data. The organizing principle of spectral analysis is the understanding of data through its decomposition according to these symmetries. In this setting, the data is viewed as an element of the vector space of complex-valued functions defined on some set X, denoted $L(X)$, upon which a group G acts by translation. Denoting this translation representation of G on $L(X)$ as ρ gives

$$(\rho(s)h)(x) = h(s^{-1}x)$$

for any $h \in L(X)$. It is often the case that G acts transitively on X in which case X is identified with the set of cosets G/H where H is the stabilizer of a fixed

"basepoint" of X. The associated representation space $L(X)$ is then isomorphic to $L(G/H)$, the subspace of right H-invariant functions of G, which as a representation space is equivalent to inducing the trivial representation of H to G.

The vector space $L(X)$ may be decomposed as an orthogonal direct sum of G-invariant subspaces, say

$$L(X) = V_0 \oplus V_1 \oplus \cdots \oplus V_r.$$

This decomposition may be dictated by representation theory (e.g. an irreducible or isotypic decomposition) or some immediately apparent symmetry of $L(X)$, or some combination of these sorts of considerations. Spectral analysis then takes the form of computing the projections of the data onto these subspaces and judging which projections are significant. The effect is to represent the data vector f as the sum $f = f_0 + \cdots + f_r$ where f_i is the projection of f in V_i. The normalized length of a given projection indicates its influence on the data; a large projection may suggest that further investigation is merited. For example, one course of action is to compute the inner product of f_i with various "naturally interpretable" functions in V_i (sometimes called *Mallows's method*). Another possible approach is to randomly generate data vectors that share some particular characteristics with the data (including the length) to determine if there is anything surprising about the size of a given projection, given these characteristics (cf. [**30**]).

This very general guiding principle encompasses various standard approaches to data analysis. A widely studied example is the analysis of time series. In this situation the goal is to analyze some function of time, say the Dow Jones average, seismograph data, or the number of babies born in New York city each day, by expanding the observed function into a sum of sines and cosines, our favorite periodic functions. The expansion obtained here is precisely the Fourier expansion, and analysis proceeds by looking for the large Fourier coefficients, *i.e.*, the large projections (see e.g. [**42**]). Computation requires a discretization and truncation of the data and in so doing, the expansion is computed as a discrete Fourier transform and is performed efficiently by the abelian FFT.

So, to reiterate, the theme is that under the assumption of a natural symmetry group for the domain of the data, group theory can be used to decompose the data as well as to indicate expansions of the data which will help in its interpretation. The use of generalized FFTs for spectral analysis is in the efficient computation of the projections. Algorithmic details can be found in the papers [**28, 32, 82, 83**] as well as the many references contained therein.

The following hopes to elucidate this approach by way of example. First a classical example due to Yates is discussed. His efficient algorithm for the analysis of data from 2^k-factorial designs is one of the early (abelian) FFT discoveries. With this in hand the nonabelian examples of (partially) ranked data and data from experimental designs are discussed. For the former, a real example is worked out in Appendix A.

2.2. Yates - An FFT for better wheat. The statistician Yates was one of the pioneers in the use and analysis of factorial designs. As a set, a 2^k-*factorial design* is simply the set of all k-tuples of $\{0, 1\}$, naturally identified with the vertices of the k-dimensional hypercube, or elements of the group $(\mathbb{Z}/2\mathbb{Z})^k$. In the context of statistical design this gives a natural way of indexing the trials of an experiment which depends on k *factors*, each of which may be set at a *high* or *low* level.

Data on such a design is simply a complex-valued function defined on $\{0,1\}^k$. Often an experiment is run repeatedly at the various combinations of low and high levels for the different treatments, and ultimately, the analysis is performed on the function f whose value at a given $w \in \{0,1\}^k$ is the average of all the readings at the treatment w. The goal is to attempt to understand how the different treatment levels interact to affect the yield. This is the so-called *factor analysis* for the experiment.

For example, consider the following imagined scenario and accompanying toy data set for a 2^3-factorial design. A farmer is interested in factors affecting the growth of his wheat. In particular he'd like to understand how sunlight, weed killer and fertilizer affect the height of the plant. Simplifying things, the plants are exposed to all three factors at various combinations of two possible levels, high and low, denoted as + or − respectively. Each of the eight possible combinations is applied to the same number of plants and finally the average height, denoted α_{swf} for a given choice of sunlight (s), weed killer (w) and fertilizer (f), at each combination is recorded. The following table gives a possible summary of such an experiment.

s	w	f	α_{swf}
+	+	+	57
-	+	+	69
+	-	+	51
-	-	+	65
+	+	-	53
-	+	-	84
+	-	-	46
-	-	-	82

Table 1. An example of a possible data set for a 2^3-factorial design.

Thus, for example, row two in Table 1 indicates that at a low level of sunlight, and high levels of weed killer and fertilizer, the average wheat plant height was 69 centimeters.

Of interest are the various effects of the factors, both individually and in combinations. There are various quantities which seem to be worth examining. The zeroth order effect is the *grand mean* or total average height, denoted μ_{gr}. This is simply the average of all the (already averaged) heights

$$\mu_{gr} = \frac{1}{8} \sum_{(s,w,f)\in\{+,-\}^3} \alpha_{swf}.$$

This estimates the intrinsic yield of simply growing wheat, *i.e.*, by virtue of planting wheat, the average height which could be expected independent of doing anything to the plant. Next to be considered might be the "pure" first order effects: the effect of one particular factor, all other factors being held equal. In the case of sunlight this might be measured by considering the differences of the average yields at a high level of sunlight versus the average at a low level

$$\mu_S = \frac{1}{4}(\alpha_{+--} + \alpha_{+-+} + \alpha_{++-} + \alpha_{+++}) - \frac{1}{4}(\alpha_{---} + \alpha_{--+} + \alpha_{-+-} + \alpha_{-++}).$$

An example of one of the three possible second order effects considers the interaction of sunlight and weed killer. This "2 factor interaction" is measured by

$$\mu_{SW} = \frac{1}{2}\left[\frac{1}{2}(\alpha_{+++} + \alpha_{++-}) - \frac{1}{2}(\alpha_{-++} + \alpha_{-+-})\right] - \frac{1}{2}\left[\frac{1}{2}(\alpha_{+-+} + \alpha_{+--}) - \frac{1}{2}(\alpha_{--+} + \alpha_{---})\right].$$

Finally, the "3 factor interaction" is measured by

$$\mu_{SWF} = \frac{1}{2}\left[\frac{1}{2}(\alpha_{+++} - \alpha_{-++}) - \frac{1}{2}(\alpha_{+-+} - \alpha_{--+})\right] - \frac{1}{2}\left[\frac{1}{2}(\alpha_{++-} - \alpha_{-+-}) - \frac{1}{2}(\alpha_{+--} - \alpha_{---})\right].$$

The collection of effects may be coded up as the computation of the following matrix-vector multiplication

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \end{pmatrix} \cdot \begin{pmatrix} \alpha_{+++} \\ \alpha_{-++} \\ \alpha_{+-+} \\ \alpha_{--+} \\ \alpha_{++-} \\ \alpha_{-+-} \\ \alpha_{+--} \\ \alpha_{---} \end{pmatrix} = H_3 \cdot \begin{pmatrix} \alpha_{+++} \\ \alpha_{-++} \\ \alpha_{+-+} \\ \alpha_{--+} \\ \alpha_{++-} \\ \alpha_{-+-} \\ \alpha_{+--} \\ \alpha_{---} \end{pmatrix}$$

$$= \begin{pmatrix} 8\mu_{gr} \\ 4\mu_F \\ 4\mu_W \\ 4\mu_S \\ 4\mu_{SW} \\ 4\mu_{WF} \\ 4\mu_{SF} \\ 4\mu_{SWF} \end{pmatrix}$$

where H_3 denotes the matrix of $+1$'s and -1's, which is an 8×8 Hadamard matrix. As the matrix-vector multiplication makes evident, what is of interest is the projection of the data vector onto the vectors given by the rows of H_3. Group theorists may immediately recognize these rows as the character values, or equivalently the irreducible matrix coefficients, of the group $(\mathbb{Z}/2\mathbb{Z})^3$. Recall that in general, for the group $(\mathbb{Z}/2\mathbb{Z})^k$, the irreducible representations are in a one-one correspondence with the group elements. If $x \in (\mathbb{Z}/2\mathbb{Z})^k$, then define the character χ_x by $\chi_x(y) = (-1)^{x \cdot y}$ where $\cdot$ is meant to indicate the usual inner product of vectors. Consequently, under an ordering indicated by the subscripts in the data vector, the x, y entry of H_3 is given by $\chi_x(y)$. As such, each row represents a one-dimensional $(\mathbb{Z}/2\mathbb{Z})^3$-invariant subspace of $L((\mathbb{Z}/2\mathbb{Z})^3)$ and the matrix-vector product is (by definition) the Fourier transform of the data. That is, the output of the matrix-vector multiplication is the collection of values $\{\hat{\alpha}(\chi)\}$ as χ ranges over all irreducible representations of $(\mathbb{Z}/2\mathbb{Z})^k$.

Remark. Notice that when computed directly, 8^2 operations are required to compute the full analysis in our example. Analogous decompositions may be obtained for any 2^k-factorial design and in general $(2^k)^2$ operations are required to compute the analysis directly. For k large this cost is prohibitive.

Yates succeeded in finding an algorithm which is much more efficient than direct computation. More precisely, he discovered an algorithm which requires at most $3k2^k = 3 \cdot 2^k \log(2^k)$ operations [**103**].

2.3. Fourier analysis for finite groups. Section 2.2 has described the use of spectral analysis for data on the group $(\mathbb{Z}/2\mathbb{Z})^k$. From the point of view of representation theory, this has decomposed a function on the group in terms of irreducible matrix coefficients. The general spectral analysis approach performs the same sort of decomposition for data on nonabelian groups and their quotients with similar interpretative aims.

Recall that the general definition of a Fourier transform on a finite group is as follows: a (complex) **matrix representation** of a finite group G is a map ρ from G into the group of $d \times d$ invertible matrices with complex entries, $GL_d(\mathbb{C})$, such that

$$\rho(st) = \rho(s)\rho(t)$$

for every $s, t \in G$. In this case $d = d_\rho$ is called the **degree** or **dimension** of the representation ρ, and $V = \mathbb{C}^d$ is called the **representation space** of ρ. The function $\rho_{ij}(s)$ defined by considering the i, j entry of $\rho(s)$ for each s is called a **matrix coefficient**.

Two representations ρ_1 and ρ_2 are said to be **equivalent** if they differ only by a change of basis, *i.e.*, if there exists an invertible matrix A such that $\rho_1(s) = A^{-1}\rho_2(s)A$ for all $s \in G$. While 1-dimensional matrix representations are uniquely determined by their equivalence class, multidimensional representations have an infinite number of equivalent realizations.

A subspace $W \subset V = \mathbf{C}^d$ is said to be G-**invariant** if for all $s \in G$, $\rho(s)W \subset W$. The representation ρ is said to be **irreducible** if $V = \mathbf{C}^d$ has no G-invariant subspaces other than the trivial subspaces $\{0\}$ and V. Othersie, ρ is said to be **reducible**. Up to equivalence, any finite group has only a finite number of irreducible representations — in fact as many as there are conjugacy classes in the group. Irreducible representations are the fundamental building blocks of all representations of a finite group. More precisely, any representation is equivalent to the direct sum of irreducible representations, where the direct sum of two representations is the matrix direct sum of the representations.

Let $\rho^{(1)}, \ldots, \rho^{(h)}$ be a complete set of inequivalent matrix representations for G with d_i equal to the degree of $\rho^{(i)}$. Then the corresponding collection of matrix coefficients $\{\rho^{(i)}_{jk} | 1 \leq i \leq h, 1 \leq j, k \leq d_i\}$ form an orthogonal basis for the $|G|$-dimensional vector space of complex-valued functions on G, denoted $\mathbf{C}G$.

A Fourier transform for a finite group G is a change of basis from the basis of point-mass or delta functions for $\mathbf{C}G$ to a basis of irreducible matrix coefficients. Following various approaches [**18, 5, 29**] a slightly more general definition for a Fourier transform for a finite group is adopted.

DEFINITION 1 (Fourier Transform). *Let G be a finite group and f be a complex-valued function on G.*

i. *Let ρ be a matrix representation of G. Then the* **Fourier transform of f at ρ,** *denoted $\hat{f}(\rho)$ is the matrix sum.*

$$\hat{f}(\rho) = \sum_{s \in G} f(s)\rho(s). \tag{1}$$

Similarly, define the **Fourier transform of f at the matrix coefficient** ρ_{jk}, *denoted* $\hat{f}(\rho_{jk})$ *as the scalar sum,*

$$\hat{f}(\rho_{jk}) = \sum_{s \in G} f(s)\rho_{jk}(s). \tag{2}$$

ii. *Let* $\mathcal{R}$ *be a set of matrix representations of* G. *Then the* **Fourier transform of** f **on** $\mathcal{R}$ *is the set of* $|\mathcal{R}|$ *(matrix) Fourier transforms of* f *at the representations in* $\mathcal{R}$, *or equivalently, the* $\sum_{\rho \in \mathcal{R}} d_\rho^2$ *scalar quantities given by the Fourier transforms of* f *at the matrix coefficients of the representations in* $\mathcal{R}$.

As in the case of the 2^k-factorial design, the Fourier transform of a function at a complete set of irreducible representations re-expresses the function in terms of a basis of matrix coefficients. This is effectively the content of the Fourier inversion formula,

$$f(s) = \frac{1}{|G|} \sum_{\rho \in \mathcal{R}} d_\rho \text{trace}\left(\hat{f}(\rho)\rho(s^{-1})\right) \tag{3}$$

where the sum is over a complete set of irreducible matrix representations for G. **Fast Fourier transforms** or **FFTs** are algorithms for computing Fourier transforms efficiently.

2.4. Spectral analysis of full and partially ranked data. To date, possibly the most successful application of the techniques of spectral analysis for a finite nonabelian symmetry group have come from Diaconis's use of this technique for the analysis of ranked data. A full example is worked out in complete detail in [25], while other examples as well as many other kinds of applications can be found in the book [26]. The following is intended to serve as an introduction to these ideas.

To explain the setting, respondents are given a list of items and asked to rank some subset of them in terms of preference. The requested ranking may be "full" in the sense that the respondent is asked to reorder the entire list linearly, or "partial" meaning that perhaps only a subset is to be chosen for ranking or a collection of subsets, e.g. of 10 items, choose your top 2 favorites (with no internal ordering) and then your next 3 (again with no internal ordering).

For full ranking of n items, respondents reply with a choice of a permutation of the originally ordered set. For example, imagine the following scenario. A movie studio is interested in current viewing trends. Respondents are presented with the list of movies: (1) *Bridges of Madison County*, (2) *Terminator*, (3) *The Piano*, (4) *Pulp Fiction*, and (5) *Unforgiven* and asked to rank them in order of preference. A possible (and personal!) response might be

Movies		*Preference Order*
(1) *Bridges of Madison County*		(4) *Pulp Fiction*
(2) *Terminator*	*D.R.*	(5) *Unforgiven*
(3) *The Piano*	——	(2) *Terminator*
(4) *Pulp Fiction*		(1) *Bridges of Madison County*
(5) *Unforgiven*		(3) *The Piano*

equivalent to a choice of the permutation 43512 (in one-line notation). If many people are asked then a function $f : S_5 \longrightarrow \mathbb{Z}$, is determined by

$$f(\pi) = \text{the number of respondents with preference order } \pi.$$

There are natural statistics to compute from such data. The first thing to be computed is the mean, or average response. This is precisely the projection of the data onto the one-dimensional space of constant functions, and is given by the constant vector, all of whose entries are equal to $\frac{1}{|S_5|}\sum_{\pi \in S_5} f(\pi)$. Restated in terms of representation theory, this is essentially the Fourier transform of the data at the trivial representation. Next, a "first order summary" of the data is obtained by simply counting how many respondents ranked movie i in position j. Notice that this is precisely the content of the Fourier transform of the data at the defining representation of S_5. That is, define

$$\rho_{ij}(\sigma) = \begin{cases} 1 & \text{if } \sigma(i) = j \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$\begin{aligned} \hat{f}(\rho_{ij}) = \langle f, \rho_{ij} \rangle &= \sum_{\sigma \in S_5} f(\sigma)\rho_{ij}(\sigma) \\ &= \text{the number of respondents ranking movie } i \text{ in position } j. \end{aligned}$$

Of course, $\rho_{ij}(\sigma)$ is just the i,j-entry of the matrix $\rho(\sigma)$, the representation of S_5 assigning the corresponding permutation matrix to σ, and the first order analysis consists of computing the Fourier transform $\hat{f}(\rho)$.

Similarly, "higher order summaries" can be obtained by computing Fourier transforms at other representations of S_5, or in the general situation, S_n. A higher order effect attempts to account for interactions in the data. For example, since Clint Eastwood stars in both *Unforgiven* and *Bridges* it could be expected that these movies might often be ranked near each other, thereby causing permutations which mapped these indices to adjacent indices to be chosen more frequently. Similarly, *Unforgiven*, *Terminator* and *Pulp Fiction* are action movies, while *Piano* and *Bridges* are romances, so some sort of clustering which reflects these common characteristics is to be expected. Relevant functions for these higher order projections would be

$$\rho_{\{i,j\},\{k,l\}}(\sigma) = \begin{cases} 1 & \text{if } \sigma(\{i,j\}) = \{k,l\} \\ 0 & \text{otherwise} \end{cases}$$

or their "ordered" cousins

$$\rho_{(i,j),(k,l)}(\sigma) = \begin{cases} 1 & \text{if } \sigma(i,j) = (k,l) \\ 0 & \text{otherwise.} \end{cases}$$

Thus, $\hat{f}(\rho_{\{ij\},\{k,l\}})$ records the number of people ranking movies i and j in positions k and l where order isn't important and $\hat{f}(\rho_{(i,j),(k,l)})$ the number where order *is* important. Implicit here is the computation of Fourier transforms for functions on the symmetric group at the well-known reducible representations given by actions on Young tableaux. The following cursory description of the theory will expedite the remaining spectral analysis discussion.

2.5. Representation theory of the symmetric group and ranked data. The relevant combinatorial object for spectral analysis of ranked or partially ranked data is the **Young tableau**, named after the British mathematician Alfred Young. To define this, let $\lambda \vdash n$ be a partition of n. Thus, $\lambda = (\lambda_1, \ldots, \lambda_h)$ where $\lambda_1 \geq \cdots \geq \lambda_h > 0$ with $\sum_i \lambda_i = n$. In this case $h = h(\lambda)$ is called the **height** of λ. A **Ferrers** or **Young diagram** of shape λ is a left-justified array of square boxes with λ_i boxes in row i.

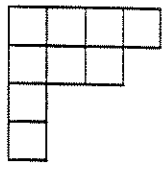

Figure 1: The Young diagram for $(4, 3, 1, 1)$.

If the numbers $1, \ldots, n$ are placed without repetition in the boxes of a Young diagram of shape λ, then a **Young tableau of shape** λ is defined. Two Young tableau of shape λ are said to be (row) equivalent if they differ only by permuting the entries within a given row. An equivalence class of Young tableaux of a fixed shape is called a **tabloid** of the same shape. It is clear how a tabloid of shape λ corresponds to a particular partial ranking. A given tabloid is denoted by forming a representative Young tableau, and removing the internal vertical lines.

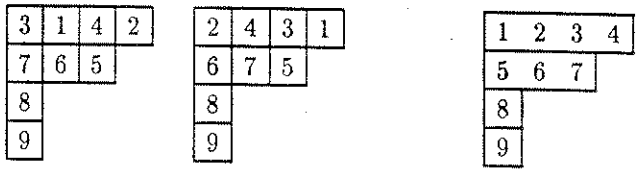

Figure 2: Two equivalent Young tableaux of shape (4,2,1,1) followed by the corresponding tabloid.

Let X^λ denote the set of tabloids of shape λ. The symmetric group on n symbols, S_n, acts transitively on X^λ by permuting the entries in the tabloids. The subgroup stabilizing a given tabloid is called a **Young subgroup**. Note that for fixed λ, each such subgroup is isomorphic to $S_{\lambda_1} \times \cdots \times S_{\lambda_h}$. Let S_λ denote the particular Young subgroup which permutes the sets $\{1, \ldots, \lambda_i\}, \ldots, \{\lambda_1 + \cdots + \lambda_{h-1} + 1, \ldots, n\}$ within themselves. This gives rise to an identification between X^λ and S_n/S_λ and a representation of S_n (by translation) on the vector space $M^\lambda = L(X^\lambda)$.

The representation theory of S_n is studied by decomposing in a systematic way the permutation representations M^λ. For present purposes it suffices to say that within each M^λ there is a uniquely determined irreducible subspace S^λ, and letting λ run through all partitions of n accounts for all the irreducible representations of S_n, without multiplicity. Sagan gives a very nice exposition of the construction [**89**]. James and Kerber provide an encyclopedic account with many references [**62**].

The discussion in Section 2.3 shows that the Fourier transforms at matrix coefficients of the representations on the reducible modules M^λ are easily interpreted. However, a spectral analysis representation rewrites the function in terms of Fourier transforms at irreducible matrix coefficients. The claim is that the Fourier transforms at the matrix coefficients for a given S^λ encode the pure interactions specified by λ.

To obtain some feeling for this connection it is perhaps easiest to work within the context of a specific example, so consider the representation $M^{(n-1,1)}$, given by the symmetric group S_n acting on Young tableaux of shape $(n-1,1)$. Any such Young tableau is determined by the entry in the second row, and thus, may be identified with the action of S_n on the standard basis $e_1, \ldots, e_n$ given by $\rho(\pi)e_i = e_{\pi(i)}$, which is the defining representation of S_n. For any fixed i, the matrix coefficients $\{\rho_{i1}, \ldots, \rho_{in}\}$ span an S_n-invariant subspace of $L(S_n)$. This follows from the fact that $(\tau\rho_{ij})(\sigma) = \rho_{ij}(\tau^{-1}\sigma)$, implying that $\tau\rho_{ij} = \rho_{i\tau(j)}$. Thus, the set of matrix coefficients $\{\rho_{i1}, \ldots, \rho_{in}\}$ do themselves span a copy of $M^{(n-1,1)}$, thereby providing n easily identified isomorphic copies of the space $M^{(n-1,1)}$. The representation space $M^{(n-1,1)}$ decomposes as

$$M^{(n-1,1)} = S^{(n)} \oplus S^{(n-1,1)}$$

where $S^{(n)}$ denotes the trivial representation, spanned by the subspace of vectors with constant coordinates, while $S^{(n-1,1)}$ denotes its $n-1$-dimensional orthogonal irreducible complement of those vectors whose coordinates sum to zero. These copies of $M^{(n-1,1)}$ are not mutually orthogonal. For example, each contains the same copy of the trivial representation. Also, notice that any one set of first order statistics $\{\hat{f}(\rho_{i1}), \ldots, \hat{f}(\rho_{in})\}$ is determined by knowing all the others. In fact, the matrix coefficients span a space of dimension $(n-1)^2+1$. Herein lies the connection between the Fourier transform $\hat{f}(\rho)$ and the decomposition of the original data vector. The entire vector space $L(S_n)$ has a particular decomposition given by representation theory, the so-called *isotypic decomposition* which can be written as

$$L(S_n) = \oplus_{\lambda \vdash n} I^\lambda. \tag{4}$$

Each subspace I^λ is equivalent to d_λ copies of S^λ. The space of matrix coefficients ρ_{ij} span a subspace of $L(S_n)$ that has an S_n-irreducible decomposition isomorphic to $S^{(n)} \oplus (n-1)S^{(n-1,1)}$ and the computation of $\hat{f}(\rho)$ is equivalent to computing the projection of f onto the trivial representation, as well as the isotypic component of $L(S_n)$ which corresponds to the irreducible representation $S^{(n-1,1)}$, denoted as $I^{(n-1,1)}$ in (4). These projections are the Fourier transforms at the corresponding irreducible representations and in this sense, the projection onto $I^{(n-1,1)}$ encodes the pure first order information about f.

A similar argument holds true for the higher order statistics as well. Thus, the summary is, that for each partition λ of n, there is a permutation representation M^λ. The matrix coefficients do themselves give a representation and the Fourier transform of the data computes the projection of the data onto this invariant subspace. In the natural basis of the representation, the corresponding Fourier transform at this basis computes certain frequency counts, but this information is both coarse and redundant. Obtaining the "pure higher order effect" (as represented by λ), is equivalent to the computation of the projection of the data onto the S^λ-isotypic, which, in turn, is the same as computing the Fourier transform of the data at the irreducible representation corresponding to λ.

Hopefully, this gives a flavor of the spectral analysis approach for ranked data. For a full analysis of a data set on S_5 see [**25**]. For other applications of spectral analysis for the symmetric group see [**4**]. The situation for partially ranked data is similar. In Appendix A an explicit example is considered.

2.6. Data on balanced incomplete block designs. In addition to ranked data, another large class of potential uses of spectral analysis comes from analyzing data on experimental designs, one example of which is the 2^k-factorial design discussed in Section 2.2. While factorial designs explicitly compare all combinations of factors, for various reasons this is not always practical or even useful. More generally, a variety of combinatorial structures are used to compare the effects of many combinations of certain controlled factors on the yield of a particular experiment. Designs are used to schedule the comparisons, hopefully in such a way that the various factor interactions can be disentangled well enough to determine the effects of the different factors (both together and separately) on the experiment.

Section 2.2 showed how Fourier or spectral analysis could be used to understand data on the 2^k-factorial design. The goal of this section is to show how general spectral analysis can be used to analyze data on another class of designs, so-called balanced incomplete block designs. Classically, such data is analyzed by the use of analysis of variance (ANOVA) for linear models (cf. [**91**]). The following section summarizes some recent work [**2**] which shows that the classical analysis often can be subsumed by a spectral analysis approach.

To begin, some of the notation and terminology of design theory must be recalled. Let $\mathcal{D}$ be any **incidence structure**. Thus, $\mathcal{P}$ and $\mathcal{B}$ are any sets with $\mathcal{P} \cap \mathcal{B} = \emptyset$ and $\mathcal{D} \subseteq \mathcal{P} \times \mathcal{B}$. The elements of $\mathcal{P}$ are called **points** or **treatments** and the elements of $\mathcal{B}$ are called **lines** or **blocks** and the elements of $\mathcal{D}$ are called **flags** or **plots**. If $(p, B) \in \mathcal{D}$ then p is said to be **incident** with B and (p, B) is called a **point-block pair**.

Example: k-sets of a v-set. For illustration it is useful to carry along an example. For this, consider the design given by all k-sets of a v-set. The blocks are all the subsets of size k within $\{1, \ldots, v\}$ and point-block pairs are of the form (i, x) where $i \in x \subset \{1, \ldots, v\}$ and $|x| = k$. As such, point-block pairs can also be represented as pairs (i, y) where $y \subset \{1, \ldots, v\}$ with $|y| = k - 1$ and $i \notin y$.

The primary example of interest here is that of a **balanced incomplete block design** (BIBD). This incidence structure has the additional properties:

(1) *All blocks are incident with a constant number, k of points;*
(2) *All points are incident with a fixed number r of blocks and*
(3) *Every pair of points is incident with an equal number λ of blocks.*

Properties (1) and (2) alone define a **tactical configuration**. More generally, condition (3) can be replaced by the analogous condition on subsets of t points, thereby defining a **t-design**. So, BIBDs are also sometimes called 2-designs. A design is **symmetric** if $k = r$. It is clear that the k-sets of a v-set comprise a BIBD.

An **automorphism** of $\mathcal{D}$ is any 1:1 mapping of $\mathcal{P} \cup \mathcal{B}$ to itself that preserves the incidence relation. The group of automorphisms of $\mathcal{D}$ is denoted $Aut(\mathcal{D})$. It is assumed throughout that all incidence structures are finite. Data on an incidence structure $\mathcal{D}$ is thus an element of $L(\mathcal{D})$, the vector space of complex-valued functions

on the incidence relation $\mathcal{D}$. The action of the group $Aut(\mathcal{D})$ on D thus gives rise to a representation of $Aut(\mathcal{D})$ on $L(\mathcal{D})$.

Example: k-sets of a v-set. The group S_n acts as automorphisms of this design, serving to mix up the labels on the treatments. Thus, representing a point-block pair as $(j, \{i_1, \ldots, i_{k-1}\})$ for $j \notin \{i_1, \ldots, i_{k-1}\})$, then for $\pi \in S_v$ define $\pi(j, \{i_1, \ldots, i_{k-1}\}) = (\pi(j), \{\pi(i_1), \ldots, \pi(i_{k-1})\})$. In the notation of Section 2.5 The corresponding translation representation is recognized as $M^{(v-k+1,k-1,1)}$.

Examples of this design can arise in "tasting" data. A classical data set of Calvin [**11**] analyzes the effect of different levels of vanilla in the manufacture of vanilla ice cream. Respondents are asked to taste the different types of vanilla ice cream and rate them, assigning a number from 1 to 5. While it may seem natural to have each person taste all flavors for best comparison, there is a problem with this approach: tasters can suffer from taste fatigue; after some time, everything tastes the same. To combat this, tasters are asked to taste only a subset of the flavors, in particular, a subset of size four of the six possible flavors. The responses give data on 4-sets of a 6-set. A (possibly) more important example comes from medical imaging and the subjective rating of images. In such examples, radiologists may be asked to rate the quality of images of a given object obtained under different image reconstruction algorithms [**22**]. They too may suffer from "taste fatigue" if they view too many pictures.

Standard techniques (see e.g. [**91**]) for analyzing such data amount to computing the projections of the data onto subspaces spanned by sets of functions suggested by the combinatorial structure of the design. This is explained as follows.

Define **pure point functions** $\{\tau_p\}_{p \in \mathcal{P}} \subset L(\mathcal{D})$ by

$$\tau_p(p', B) = \begin{cases} 1 & p = p' \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

for any flag (p, B).

Similarly, define the **pure block functions** $\{\beta_B\}_{B \in \mathcal{B}} \subset L(\mathcal{D})$ by

$$\beta_B(p, B') = \begin{cases} 1 & B = B' \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

for any flag (p, B). Finally, also define the **adjusted treatment functions** α_p as

$$\alpha_p = \tau_p - \frac{1}{k} \sum_{B \ni p} \beta_B \tag{7}$$

where recall that k is the number of points incident with any block.

Let $f \in L(\mathcal{D})$, so $f : \{(p, B) | p \in B\} \longrightarrow \mathbf{C}$. Consider the effect of projecting f onto the subspace spanned by the subsets $\{\tau_p\}_p$, $\{\beta_B\}_B$ and $\{\alpha_p\}_p$ respectively. The $\{\tau_p\}_{p \in P}$ are orthogonal and projection onto any particular τ_p is simply

$$\left(\frac{1}{r} \sum_{B \ni p} f(p, B) \right) \tau_p.$$

Thus, any vector in the span of the pure point functions is a function that only depends on the point, or treatment, with no dependence on the block containing

that treatment. Projection onto this space is creates the function

$$\sum_p \left(\frac{1}{r} \sum_{B \ni p} f(p, B) \right) \tau_p$$

whose value at a given point-block is the average value of the yield at that point. Similarly, the span of the pure block functions are those functions which only depend on the block and not the individual treatments of which it is comprised. Projection onto this space creates a function whose value at a given point-block is the average value of the yield on that block. These projections are meant to measure the individual effect of a given point and block respectively. The effect of projecting f onto the span of the α_p is to remove the effect of a given block from the yield at the treatment within the block.

The span of the τ_p is naturally thought of as a "Treatment Space". Similarly, it is natural to call the span of the β_B the "Block Space" or as it is often called the "blocks-ignoring-treatment" subspace. Notice that this subspace contains the "Mean Space", given by the 1-dimensional subspace V_0 of constant functions, now familiar as the space whose projection gives the mean response. Thus,

$$span\{\beta_B | B \in \mathcal{B}\} = V_0 \oplus V_{bl.ig.tr}.$$

Finally, the subspace spanned by the α_p is thought of as a "Treatment Space" and classically is sometimes called the "treatments-eliminating-blocks" subspace,

$$span\{\alpha_p | p \in \mathcal{P}\} = V_{tr.el.bl.}.$$

Standard analysis of data on a design as a linear model simply computes the projection of the data onto each of these subspaces and compares and analyzes the lengths of the projections. What remains after these projections are analyzed is the length of the projection onto an "Error Space", which usually lacks any deeper interpretation. Thus, we write

$$L(\mathcal{D}) = V_0 \oplus V_{bl.ig.tr} \oplus V_{tr.el.bl.} \oplus V_{error}.$$

The tie to group theory is that each of these subspaces is $Aut(\mathcal{D})$-invariant. (The Error Space can be chosen as such.) A spectral analysis point of view suggests that finer group-invariant structure should be investigated. This is illustrated in the context of the example of k-sets of a v-set, as studied in [2].

Example. The k-sets of a v-set. In order to simplify the analysis assume $v \geq 2k$ and in particular, consider the case of the design of all 4-sets of a v-set where $v \geq 8$. Thus, the parameters become

$$k = 4, \quad v \geq 8, \quad b = \binom{v}{4}, \quad N = 4 \cdot \binom{v}{4}.$$

Thus, the set of flags for this design are the pairs $\{x, i\}$ where x is a 4-set of $[v] = \{1, \ldots, v\}$ and $i \in x$. The symmetric group S_v acts transitively on the set of flags. The spectral analysis approach considers the corresponding translation representation and looks to interpret an irreducible decomposition.

In the standard notation of the representation theory of the symmetric group (cf. Section 2.5) the action on the design of 4-sets is $M^{(v-4,3,1)}$ (note that the

stabilizer of a point-subset pair fixes the distinguished point, but is free to permute the remaining 3 elements arbitrarily). This has decomposition

$$M^{(v-4,3,1)} = S^{(v)} \oplus S^{(v-1,1)} \oplus 2S^{(v-2,2)} \oplus 2S^{(v-3,3)} \oplus S^{(v-3,2,1)} \oplus S^{(v-4,4)} \oplus S^{(v-4,3,1)} \quad (8)$$

with dimension decomposition

$$4\binom{v}{4} = 1 + 2(v-1) + 2(\binom{v}{2} - v) + 2(\binom{v}{3} - \binom{v}{2}) + \frac{v(v-2)(v-4)}{3} + \binom{v}{4} - \binom{v}{3} + \frac{v(v-1)(v-3)(v-6))}{8} \quad (9)$$

and subspace identifications

$$\begin{array}{rcl} \text{Mean Space} & \Leftrightarrow & S^{(v)} \\ \text{Treatment Space} & \Leftrightarrow & S^{(v-1,1)} \\ \text{Block Space} & \Leftrightarrow & S^{(v-1,1)} \oplus S^{(v-2,2)} \oplus S^{v-3,3} \oplus S^{(v-4,4)} \\ \text{Error Space} & \Leftrightarrow & S^{(v-4,3,1)} \end{array} \quad (10)$$

For several of the subspaces the choices are not canonical. Thus, just for definiteness, notice that the Treatment Space is precisely the subspace spanned by the functions $\tau_i = \sum_{x \ni i} \delta_{x,i}$ (with the mean subtracted), and the Block Space is spanned by the functions $\beta_x = \sum_{i \in x} \delta_{x,i}$. (Here $\delta_{x,i}$ is the function equal to one at the point block pair (x, i) and zero elsewhere.) It is not too hard to check that with these definitions the two copies of $S^{(v-1,1)}$ are in fact orthogonal. The irreducible decomposition of the Block Space proceeds according to the usual analysis of k-sets of a v-set [**26**]. Thus, what remains are the "new" spaces (which classically are subspaces of the error space). These are the direct summands,

$$S^{(v-2,2)} \oplus S^{(v-2,1,1)} \oplus S^{(v-3,3)} \oplus S^{(v-3,2,1)}. \quad (11)$$

To begin the discussion, first note that the subspace $S^{(v-2,1,1)}$ is what is called the *adjusted correlation space* in [**11**]. This deserves some explanation: From the point of view of the symmetric group, this is intrinsically an ordered space, *i.e.*, it is the "pure" effect of functions defined on ordered pairs in an v-set. However, block designs are intrinsically unordered. The new insight is this. As the above shows, there are **two** second order correction pieces. The first $S^{(v-2,2)}$ is a piece added due to the presence of i and j in the block, no matter what treatment is observed and the second is the effect due to j being present when i is observed. Note that this is intrinsically an ordered thing and it is $S^{(v-2,1,1)}$.

The third order pieces $S^{(v-3,3)} \oplus S^{(v-3,2,1)}$ have the same flavor. The first is an additive piece due to unordered triples. The second is the additional effect of the triple when i is observed.

Remark. Other aspects of this analysis are discussed by Fortini [**40**] whose work is summarized by Diaconis in [**24**] and in the Discussion following [**94**].

2.7. FFTs for nonuniform frequencies. A final example of the use of techniques of generalized FFTs for data analysis comes from the application of fast discrete monomial transforms to the efficient evaluation of DFTs at roots of unity,

other than the usual equispaced ones. These so-called *nonuniform DFTs*, are computations of the form

$$\hat{f}(j) = \sum_{k=0}^{N-1} f_k e^{2\pi i k x_j} \tag{12}$$

for $x_0, \ldots, x_{N-1}$ an arbitrary sequence of points on the real line. In the case in which $x_j = j/N$ then we have the usual (uniform) DFT.

Nonuniform sample points arise in many situations, among which are a variety of techniques for the efficient, approximate reconstruction of two-dimensional magnetic resonance imaging (MRI). The result of an MR scan is to effectively compute the two-dimensional Fourier transform of a density function in a given anatomical slice (cf. [77] for an introduction to these ideas). If all the Fourier transforms are computed, then the density can be reconstructed exactly via the usual (abelian) two-dimensional Fourier inversion. Some attempts to speed the reconstruction instead compute curves through Fourier space (*i.e.*, parametrized values $\hat{f}(x, a(x))$ for some path a) which then provide data for approximate reconstructions which amount to the computation of nonuniform DFTs (see eg. [104]).

Recent work of Driscoll, Healy and Rockmore give an exact $O(N \log^2 N)$ algorithm for computing nonuniform DFTs [32]. Their technique relies on finding a factorization of the *transpose* computation, the *discrete monomial transform* $\sum_k f_k (e^{2\pi i x_k})^j$, which is good for any collection of polynomials satisfying a three-term recurrence. Issues of stability are treated in [84]. Fast approximate multipole methods (requiring $N \log N + N \log(1/\epsilon)$ operations for a desired accuracy ϵ) are due to Dutt and Rokhlin [33].

3. Filtering - Applied convolution

3.1. Cooley-Tukey - An FFT for digital signal processing. The terminology and context of signal processing is useful to explain and motivate filtering. It is with this purpose in mind that the following snapshot of the Cooley-Tukey FFT is presented.

Cooley and Tukey were interested in the analysis of time series. Loosely speaking, this is data - now called the signal - which is indexed by time. The particular application that Cooley and Tukey had in mind early in the 1960's was the analysis of seismic data. At this time, a nuclear test with the (then) USSR was under negotiation. The USSR was balking at the notion of site visits, so it was necessary that there be some way of remotely confirming compliance. The prevailing idea was to surround the Soviet Union with many sensors in order to monitor seismic activity. Nuclear detonations could then be detected by particular structure in the Fourier transforms of the collection of time series. In general, long time series were obtained and fast algorithms were mandatory for a successful analysis (cf. [19, 20] for a thorough history).

If the continuous signal were a periodic function on the real line with only a finite expansion in terms of sines and cosines (*i.e.*, "band-limited") then it is possible to compute the Fourier expansion exactly from only a finite number of values of the function - also called *samples*. It suffices to assume that f is periodic of period one, so that the additional assumption of a finite Fourier expansion implies

that f may be written as

$$f(t) = \sum_{k=0}^{N-1} \hat{f}(k) e^{2\pi i k t}$$

for $0 \leq t < 1$. The number N is called the **bandwidth** of f. The **Fourier coefficients** $\hat{f}(k)$ are given by the formula

$$\hat{f}(k) = \int_0^1 f(t) e^{-2\pi i k t} dt. \tag{13}$$

The assumption of finite bandwidth permits the the Fourier coefficients to be computed in closed form by finite sums discretizing the integrals (13). Such results go by the name of quadrature or sampling theorems. In this case, the rule is that if f is band-limited with bandwidth N, then the following well-known quadrature rule holds:

$$\hat{f}(k) = \frac{1}{N} \sum_{j=0}^{N-1} f(\frac{j}{N}) e^{-2\pi i k j / N}. \tag{14}$$

Thus, the signal may be recovered from its finite collection of equispaced samples $\{f(\frac{j}{N}) | 0 \leq j < N\}$, in the sense that this information is sufficient to compute its Fourier coefficients.

The collection of Fourier coefficients $\{\hat{f}(0), \ldots, \hat{f}(N-1)\}$ is called the **discrete Fourier transform** (DFT) of f. If computed directly, each coefficient would require N operations giving a total of N^2 operations to compute the DFT. As N becomes large this cost quickly becomes prohibitive. Cooley and Tukey derived and implemented an algorithm which given a prime factorization of $N = p_1 \cdots p_r$, computes the DFT in $O(N \sum_i p_i)$ operations [**21**]. If each $p_i = 2$, this is $O(N \log_2 N)$ operations. This algorithm, together with many variants as well as different approaches to computing the DFT, yield a family of techniques (see e.g. [**36, 99**]) which for any N give an $O(N \log N)$ algorithm for computing the DFT (cf. [**3, 27**]). This speed-up has had tremendous significance, effectively making digital signal processing a reality.

The discussions of the previous sections have really only emphasizes the spectral analysis uses of the FFT. To steer the discussion towards filtering, it is important to emphasize that the above introduces many simplifying approximations which can have a serious effect on any actual data analysis. For example, precise application of the quadrature rule (14) is based on a priori exact knowledge of the signal's bandwidth. In practice, such finely delineated signals are difficult to find.

Filters are used in order to correct for or mitigate the effect of "impurities" in an actual signal. Loosely speaking, a filter provides a means of recovering an object of interest possibly hidden in some confounding medium. A signal is sent from a source, and in transmission may be distorted due to "noise" in the medium. Thus, the measured signal may not be precisely what was sent, or as Tukey has said "That which is measured is not the truth". The purpose of a filter in this case is (to hopefully) bring the data closer to the truth. As above, this may also take on the role of identifying and matching certain characteristics within the data.

These are the goals of filtering, the former sometimes also called smoothing, while the latter is often referred to as matched filtering. In both cases the classical FFT can be useful as an efficient algorithm for effecting a filter, and there are natural nonabelian generalizations worth considering.

What follows is meant to be a brief tutorial in filtering for the uninitiated group theorist, the goal of which is to serve as enough of an introduction to interest a subset of the intended audience. There is a vast literature attached to the subject of filtering, although most sources are written primarily in the language of engineering. In this setting, a standard excellent reference is [**85**]. The recent book of Steiglitz [**95**] is a beautiful introduction to filtering from the point of view of computer music. Of primary relevance for this section is the work of Holmes whose papers [**59, 60**] do a great job of giving a group theoretic context to many signal processing constructions, and should be considered required reading for anyone interested in pursuing related research. In some ways, the following can be viewed as an introduction to these papers.

3.2. Convolution. Filtering can be viewed as a specific application of convolution. In the finite case, if $f, h \in L(G)$, then the **convolution** of f with h, denoted $f \star h$, is defined as

$$f \star h(x) = \sum_{y \in G} f(xy^{-1})h(y). \tag{15}$$

If instead G is continuous and compact, the summation is replaced by an integral against Haar measure. Notice that in the finite group case, the definition of convolution is precisely the definition of multiplication in the group algebra with respect to expansion in a basis of δ-functions, *i.e.*, $f \mapsto \sum_x f(x)\delta_x$ and $h \mapsto \sum_x h(x)\delta_x$ then $f \star h \mapsto \sum_x f \star h(x)\delta_x$.

The Fourier transform turns convolution into multiplication. For finite groups this means that for any irreducible representation ρ of G,

$$\widehat{f \star h}(\rho) = \widehat{f}(\rho) \cdot \widehat{h}(\rho). \tag{16}$$

For compact groups an analogous formulation can be made (cf. [**70**]). If computed directly, convolution would require $|G|^2$ operations. However, efficient forward and inverse Fourier transform algorithms make possible the construction of fast group convolution algorithms according to the following scheme:

(1) For all irreducible representations ρ compute $\widehat{f}(\rho)$ and $\widehat{h}(\rho)$
(2) For all irreducible representations ρ compute all products $\widehat{f}(\rho)\widehat{h}(\rho)$
(3) Apply Fourier inversion to the collection of matrices $\{\widehat{f}(\rho)\widehat{h}(\rho)\}$.

To count the number of operations required by the above method, let $T(G)$ is the number of operations required to compute a Fourier transform (or its inverse) on G. Then step 1 requires at most $2T(G)$ operations, step 2 requires at most $\sum_\rho d_\rho^3$ operations (at most n^3 operations are required to multiply two $n \times n$ matrices) and finally step 3 requires at most $T(G)$ operations. Using the the fact that $\sum_\rho d_\rho^2 = |G|$ and $d_\rho \leq |G|^{1/2}$, we see that the above is an algorithm which computes convolution in at most $3T(G) + |G|^{3/2}$ operations. Clausen has shown that $T(G) \leq |G|^{1.44}$ for any G (cf. [**10**]), so that the above algorithm has a complexity of at most $3|G|^{1.44} + |G|^{3/2}$. In the particular case of abelian groups (all irreducible representations are one-dimensional and $T(G) = O(|G| \log |G|)$, this gives an $O(|G| \log |G|)$ convolution algorithm.

3.3. Smoothing. It is perhaps best to start with the classical theory, in which smoothing aids in the analysis of time series. In this context smoothing can be effected in either the "time" or "frequency" domain, meaning that it is specified

either in terms of time or frequency. The more general context specifies the filter either on the level of the group or its dual (*i.e.*, irreducible matrix coefficients).

3.3.1. *Time smoothing.* It is instructive to think of smoothing in the time domain as a "moving average", possibly designed to account for the fact that measurements of a signal or data may be correlated in time. A typical example of smoothing by a moving average is the following. Assume $f : \{0, \ldots, N-1\} \longrightarrow \mathbf{C}$ is a "signal", with the indexing meant to indicate time. Suppose that $f = y + n$ where y is the "true" signal and n represents some noise, or perturbation of the truth. A smoothed version of f, denoted f_s might have the form

$$f_s(x) = \frac{1}{4}f(x-1) + \frac{1}{2}f(x) + \frac{1}{4}f(x+1) \tag{17}$$

with $x \pm 1$ evaluated modulo N. Figure 3 gives a pictorial representation of an example.

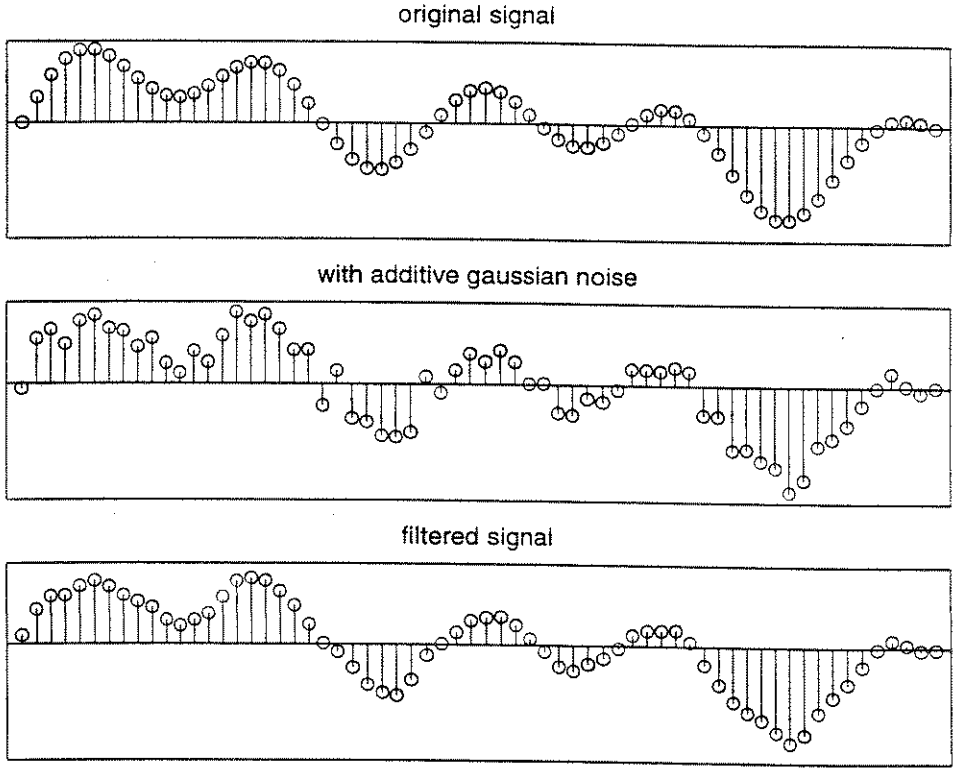

Figure 3: From top to bottom: (1) the true signal y; (2) the received signal f with white noise n added and (3) the smoothed signal f_s.

As the definition (15) shows, f_s is the convolution of f with a fixed mask m defined by

$$m(y) = \begin{cases} \frac{1}{2} & \text{if } y = 0 \\ \frac{1}{4} & \text{if } y = 1, -1 \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Hence, $f_s = f \star m$. Notice that this is precisely a discrete analogue of the (possibly) more familiar smoothing of a function on the line via convolution with a bump function.

A generalization of this idea to other groups is immediate, although to date it has not found any application. The example (17) could be formulated as follows: Let $T \subset G$ be a symmetric set of generators and let $f \in L(G)$. Smooth f according to the formula $f_s(x) = \sum_{a \in T} w_a f(ax)$ where $w_a \geq 0$ and $\sum_{a \in T} w_a = 1$. This can be interpreted as viewing the data as coming from the Cayley graph of G generated by T, so that the filter corrects a given value according to possible influence by neighboring values.

3.3.2. *Frequency smoothing.* The simplest (although often unrealistic) model to consider here is one in which a signal is corrupted by additive noise of known spectral content (frequency). By smoothing in the frequency domain the noise can be removed. For example, suppose that the signal of interest is known to only have $\hat{f}(j) \neq 0$ for $j \in A \subset \{0, \ldots, N-1\}$. By using the inverse Fourier transform, a function m with the following Fourier transform can be synthesized.

$$\hat{m}(j) = \begin{cases} 1 & \text{if } j \in A \\ 0 & \text{if } j \notin A \end{cases} \tag{19}$$

Thus, by (15) the effect of convolving with m is to remove all Fourier coefficients of frequency j for $j \notin A$ and allow all others to pass through unaffected. Such a filter is sometimes called a *bandpass* filter. Common cases which occur have $A = \{-k, \ldots, 0, k\}$ (lowpass filter) or $A = \{\frac{N}{2} - k, \ldots, \frac{N}{2} + k\}$ (highpass filter) with all indices considered mod N. The generalization to the nonabelian setting is clear.

3.4. Statistical signal processing. The scenarios of 3.3.2 come from a highly simplified model in which the noise is known exactly. In practice, what happens is that the corrupting noise is unknown and various models for the noise are assumed - usually as the output of some sort of random process. The effect of this is that when the signal is finally sampled and ready for processing, only statistical statements can be made regarding the effect of removing noise. This is the domain of statistical signal processing, a subject at the heart of modern technology. See [68] for an introduction to the subject, as well as [87] for an introduction to probability and statistics from a signal processing point of view.

Thus, this is a situation in which the data $f = s + e$ for f, s, e random vectors of length n (*i.e.*, vectors whose entries f_i, s_i, and e_i, $i = 0, \ldots, n-1$ are random variables). Section 3.3 describes filtering as convolution against a fixed function, which can either be specified according to its values on group elements or its Fourier transform. In either case, we may view filtering as the application of an $n \times n$ matrix H to the data f in order to arrive at an estimate $H \cdot f$ of the signal s.

Thus, the question arises, what to use for the filter H? Well, certainly if there were a *best* choice, then that would be the one to use, so how to determine the best choice? In general it would seem like a good idea to choose H so that on average, the distance (as vectors) between the estimate and the signal is minimized. Such an optimal choice is called a *Weiner filter*. A *generalized Weiner filter* is TWT^{-1} for W a Weiner filter and T any unitary matrix. Of course this is nothing but a change of basis, and has the property of maintaining the length, or equivalently the total energy, of the signal estimate. The choice of T which diagonalizes W is the matrix

whose columns are the eigenvectors of W, and this is called the *Karhunen-Loeve* transform (KLT) or equivalently, the K-L basis for W.

Given some knowledge of the expectation and covariance structure of the signal and noise, the Weiner filter can be solved for exactly in terms of the covariance matrices. Notice that in general, application of the Weiner filter will require n^2 operations, which for n large enough may be a prohibitive computational cost. Thus, at times, it may be preferable to use filters, which although not best in terms of average performance, may still have fairly good performance (*i.e.*, on average provide an estimate close to best) but are much faster (e.g., $O(n \log n)$). These are *suboptimal Weiner filters.*

The work of Karpovsky and Trachtenberg [67] and Holmes [59, 60] indicates that for some classes of stochastic signals, certain DFTs for nonabelian groups may be employed as suboptimal Weiner filters. The construction is quite simple. Suppose that the goals is to estimate some N-dimensional signal. Choose a group G of order N, and a fixed ordering of group elements $G = \{s_1, \ldots, s_N\}$ and complete set of irreducible $\mathcal{R} = \{\rho^{(1)}, \ldots, \rho^{(h)}\}$, with $\rho^{(i)}$ of dimension d_i. Since $\sum_i d_i^2 = |G|$, consider the $|G| \times |G|$ matrix $\mathcal{F}_{G,\mathcal{R}}$ with entries given by the the matrix coefficients $\rho_{jk}^{(i)}(s_l)$. The matrix $\mathcal{F}_{G,\mathcal{R}}$ has columns indexed by group elements and rows by the multi-index i, j, k. Thus, the first d_1^2 rows will contain the matrix coefficients $\rho_{jk}^{(1)}(s_l)$, the second d_2^2 the matrix coefficients $\rho_{jk}^{(2)}(s_l)$ and so on. Notice that for any $h \in L^2(G)$, written as the appropriate row vector with i^{th} coordinate $h(s_i)$, the product

$$\mathcal{F}_{G,\mathcal{R}} \cdot \begin{pmatrix} h(s_1) \\ \vdots \\ h(s_N) \end{pmatrix}$$

computes $DFT_{\mathcal{R}}(h)$, with consecutive blocks of d_i^2 entries of the product consisting of the entries in $\hat{h}(\rho^{(i)})$. The Fourier inversion formula (3) shows that $\mathcal{F}_{G,\mathcal{R}}$ is invertible.

A **group filter** (from G) is then any matrix of the form

$$\mathcal{F}_{G,\mathcal{R}}^{-1} \cdot D \cdot \mathcal{F}_{G,\mathcal{R}}. \tag{20}$$

where D is an $N \times N$ block diagonal matrix with i^{th} diagonal block an arbitrary $d_i \times d_i$ matrix. This ordering assumes that $\mathcal{F}_{G,\mathcal{R}}$ has the form described above. The filter is said to be nonabelian if G is nonabelian. The application of this sequence of operators has the following interpretation: view the data or signal f as a function on G, then application of the group filter (20) is equivalent to computing the convolution of f with the function h which has $\hat{h}(\rho^{(i)}) = D_i$, the i^{th} diagonal block of D (cf. Section 3.2.

For "good" choices of $\mathcal{R}$, that is, choices that give FFTs for G, and structured choices for D, the filters constructed in this can be evaluated efficiently. (In fact, some nonabelian groups of order N have FFT algorithms which are faster than abelian FFTs for groups of the same order! cf. [59]). The results of Karpovsky and Trachtenberg as well as Holmes show that some of these nonabelian filters can also serve as good approximants to Wiener filters for some classes of stochastic signals. This work only touches the surface of what might be possible for group filters and there is much room for both theoretical and practical improvement.

3.5. Data compression. The group filter constructions in the above section are easily adapted (by change in point of view) to a possible application of nonabelian DFTs for data compression. Again, the basic idea is quite easy to understand. A given class of signals (vectors) may be such that in general, after a specific change of basis (filtering), the new coordinates are small in magnitude or highly concentrated on just a few of the new basis vectors, or some combination of these things. In any case, the point of this intuitive description is to say that the change of basis has caused the original information (the original coefficients) to be compressed - in the sense that the representation in the new basis should require less space for storage.

The efficacy of a given compression scheme (change of basis) is measured by its effect on the covariance matrix of the signal. Again assume N-dimensional data is given by a signal f of constant (although not necessarily stationary) variance and let Σ_f denote the associated covariance matrix. Then for any $N \times N$ matrix U consider

$$\Sigma_f(U) = U^* \Sigma_f U = \Sigma_{Uf}.$$

Compression of a given matrix U is measured by the geometric mean of the diagonal of $\Sigma_f(U)$. As in the case of Weiner filtering, this is known to be minimal for U equal to the K-L transform for for Σ_f – *i.e.*, when U diagonalizes the covariance. Notice that this is equivalent to finding a basis which decorrelates the signal, that is, breaking it up into basic signals which act independently. As argued above, the problem is that a general K-L transform will not have an efficient algorithm and thus may be useful to instead use nonabelian DFTs as suboptimal K-L transforms. taking $U = \mathcal{F}_{G,\pi}$, Holmes computes several examples which indicate nonabelian groups can effect good compression [**59, 60**].

Remark. For various classes of signals no a priori model is given and instead, the scientist is forced to experimentally estimate the statistical properties of the class of interest. Certain classes of natural images, or local regions within images may have in common certain specific characteristics [**75**]. In cases like this in which a natural class is studied, it is possible to experimentally determine a K-L basis from a collection of samples and use this process signals from the collection. This has found many applications in many areas including pattern recognition [**41**] and magnetic resonance imaging [**58**]. Recent work seems to indicate that for some images, viewing them as functions of particular groups formed by wreath products. and thereby applying group filters constructed from wreath products, that good approximants to K-L bases can be obtained [**54**].

3.6. Matched filtering - pattern matching. Convolution can also be used to find translated copies of a fixed pattern in a signal. To state things a bit more precisely, suppose that $f : G \longrightarrow \mathbf{C}$, is a fixed function (pattern or template). Many tasks can be phrased as the problem of deciding upon either the absence or presence of a possibly translated version of f in an arbitrary input signal. That is. given $s(x) \in L(G)$ to determine if $s(x) = f_\tau(x) = f(x\tau)$ for some $\tau \in G$. This is usually complicated by the fact that the received signal $s(x)$ has been corrupted by some ambient noise during transmission. For example, under the assumption that the noise adds to the signal, the problem becomes to determine if either $s(x) = n(x)$ where n is a random variable defined on the group (with some specified statistical properties) or if $s(x) = f_\tau(x) + n(x)$ for some τ.

A classic example comes from the following highly simplified version of radar and sonar applications for determining the distance to some fixed target. In this case a fixed signal $f(x)$ is sent out from a transmitter towards an object. Upon hitting the target, the signal is returned to the receiver. Upon detection by the receiver, the distance to the target may be computed from the time elapsed between transmission and receipt. The receiver receives a possibly time-translated version of the original signal. A bit more precise formulation of this problem has $G = \mathbb{Z}/n\mathbb{Z}$ (time) and the original signal is a function $f : \mathbb{Z}/n\mathbb{Z} \longrightarrow \mathbf{C}$. After transmitting this signal the receiver then begins listening for any sort of translate of f, in this case $f_\tau(x) = f(x - \tau)$ for any τ. The way in which this is done is as follows:

The received signal is accepted in blocks of length n. Let $r(x)$ be one of these blocks. The idea is that translates of f can be detected by using convolution. For any f, define $\tilde{f}(x) = f(-x)$. Using the Cauchy-Schwartz inequality it is easy to see that if only translates of the uncorrupted signal were received (*i.e.*, $r = f_j$ for some j), then $f \star \tilde{r}(k)$ is largest at $k = j$. In the noisy model things are a bit more complicated. However, it is still possible to construct signals $f(x)$ such that under certain assumptions $f \star r(j)$ is large when $r = f_j$ and small otherwise. The types of signals constructed may depend on the models and modalities of signal transmission. This means of detection according to correlation coefficients (convolution) is also known as *maximum likelihood estimation* and uses the framework of information theory. For a friendly introduction to these ideas and basic references see [46].

In the context of image processing these sort of ideas can be used to find particular patterns in an image. As an example, consider Figure 4, computed by Peter Kostelec and Doug Warner. Here the goal is to determine the location of the 64-by-64 pattern, shown on the top left, in the image on the top right. Both images correspond to functions on R^2, sampled on a 256-by-256 grid, which store the grey scale value of each pixel. In a usual case the template and image (considered as functions on $\mathbb{Z}/256\mathbb{Z} \times \mathbb{Z}/256\mathbb{Z}$) would simply be convolved and the "big" coefficient would indicate the position of the template. As described above this would be accomplished by pointwise multiplication of the DFTs and inverting. It turns out that in this particular example, an additional operation is required. The DFTs are pointwise normalized by dividing by the magnitudes of the individual coefficients. Now as in the usual case, these normalized coefficients are pointwise multiplied and the resulting function is inverted, and again, the large coefficient marks the spot! (This modified convolution was presented at the 37th Annual Meeting of the American Association of Physicists in Medicine, Hynes Convention Center, Boston MA, 23-27 July 1995, by J.-Z. Wang, L.E. Reinstein, J. Hamley, and A.G. Meek, Dept. of Radiation Oncology, SUNY at Stony Brook in "A New Image Registration Method: Phase-Only Correlation".) The result of this modified convolution is shown in Figure 4. Notice that almost all coefficients are zero, except near the target region.

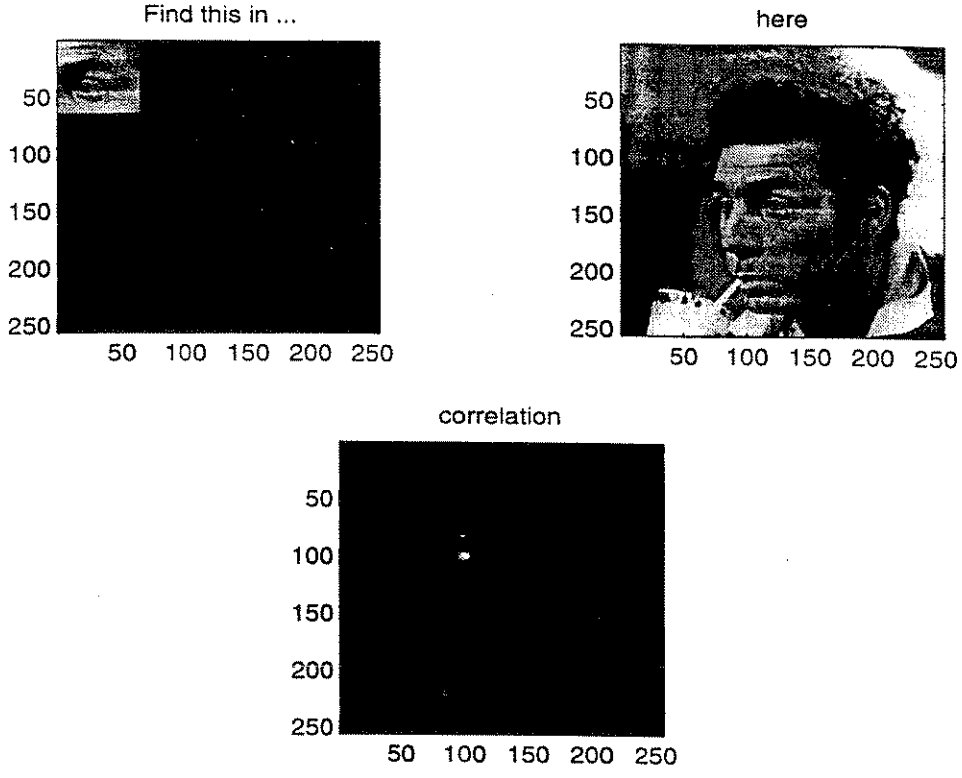

Figure 4: The search for Kramer's eye.

These ideas can be generalized to other groups and their quotients. That is, given a group or its homogeneous space, and function f (the pattern) defined on this object, it is possible to search for translates of this function via convolution, again convolving the template over the image and looking for the large coefficients. For a group G acting transitively on a set X, the convolution of a template f against another signal g is defined as the new function on X, $f \star g$ by $f \star g(x) = \sum_{\alpha \in G} f(\alpha x_0) g(\alpha^{-1} x)$ where x_0 is some fixed basepoint.

For example, there have been several attempts to use the usual permutation action of the dihedral group D_4 on a square grid to find patterns in gray-scale images (cf. [**34, 73**]). In the same spirit, recent investigations have used a variety of different actions of wreath products on a square grid to attempt to pick out rotated and translated versions of fixed patterns [**54**]. In the continuous setting, the recent discovery of an FFT on the 2-sphere [**31, 57**] makes available the possibility of matched filtering on this surface. This has potential applications in computer vision and CAD. In fact, the FFT on the 2-sphere has a wealth of possible applications and in order to highlight them, they are collected in Appendix B.

4. Adapted representations

Many of the efficient algorithms for pursuing the applications of Sections 2 and 3 depend on the use of *adapted representations* or equivalently, *adapted bases.* To put things briefly, these are bases or representations which take advantage of

certain group symmetries to simplify the form of an operator or family of operators which respect these symmetries. In addition to enabling the previously applications, these representations have other computational uses, a few of which are are outlined below.

4.1. Construction of nonabelian FFTs for finite groups. Most of the known techniques for constructing FFTs for nonabelian finite groups use irreducible representations which are adapted to a chain of subgroups. The paper [**81**] discusses this in great detail, but the following brief synopsis is included to indicate this application.

A set of matrix representations $\mathcal{R} = \{\rho_1, \ldots, \rho_h\}$ of a finite group G is said to be **H-adapted** for a subgroup $H < G$ if there is a set of matrix representations $\mathcal{S} = \{\eta_1, \ldots, \eta_r\}$ of H such that any ρ_i is equal to direct sum of η_j's when restricted to H. This definition extends directly to the notion of representations adapted for a chain of subgroups.

Adapted representations are useful for computing Fourier transforms because they permit factorization of a Fourier transform into a linear combination of Fourier transforms over a subgroup, thereby giving the first step in the development of a divide and conquer algorithm for this computation. To indicate this idea, fix a set of coset representatives Y for G/H, and irreducible representations $\mathcal{R}$ of G and $\mathcal{S}$ of H. For $f : G \longrightarrow \mathbf{C}$ and $\rho \in \mathcal{R}$, consider the following factorization of $\hat{f}(\rho)$,

$$\begin{aligned} \hat{f}(\rho) &= \sum_{x \in G} f(x)\rho(x) \\ &= \sum_{y \in Y} \sum_{z \in H} f(yz)\rho(yz) \\ &= \sum_{y \in Y} \rho(y) \sum_{z \in H} f_y(z)\rho(z) \end{aligned} \tag{21}$$

where $f_y : H \longrightarrow \mathbf{C}$ is defined by $f_y(z) = f(yz)$. Assuming that ρ is H-adapted and $\rho = \eta_{i_1} \oplus \cdots \oplus \eta_{i_m}$ (where the $\eta_{i_j} \in \mathcal{S}$) then the inner sum of (21), $\sum_{z \in H} f_y(z)\rho(z)$ can be rewritten as

$$\hat{f}_y(\eta_{i_1}) \oplus \cdots \oplus \hat{f}_y(\eta_{i_m}). \tag{22}$$

Consequently, computation of $\hat{f}(\rho)$ can be accomplished by first computing the $\hat{f}_y(\eta)$ for all $y \in Y$ and irreducible representations $\eta \in \mathcal{S}$, forming the appropriate direct sums (22) and gluing them together with the matrices $\rho(y)$. In this way computation of the Fourier transform over G is reduced to a combination of Fourier transforms over H. This is one of the fundamental ideas in the development of nonabelian FFTs and can be found in the papers [**29, 17**].

4.2. Eigenvalue and eigenvector analyses for Cayley graphs and random walks. The *adjacency matrix* of a graph X (without multiple edges) is a square matrix $A = A(X)$ with rows and columns indexed by vertices of X with entries given by

$$A_{xy} = \begin{cases} 1 & \text{if } x \text{ is connected to } y \\ 0 & \text{otherwise.} \end{cases}$$

It turns out that the spectrum of the adjacency matrix or closely related operators such as the Laplacian, encodes much information concerning the connectivity of X. For example many of the recent results on so-called expander graphs use this sort

of eigenvalue analysis in a strong way. The books [7, 76, 90] explain some of these connections and contain many references.

When $X = X(G, S)$ is a Cayley graph for a group G and generating set S, Fourier analysis can simplify the eigenvalue analysis for the associated adjacency matrix. Recall that the Cayley graph $X(G, S)$ has vertex set G and x is connected to y if and only if $y = sx$ for some $s \in S$. Then $A = A(G, S)$ is the Fourier transform of the characteristic function of S, denoted δ_S, at the left regular representation of G, ρ_{reg}, with respect to the basis of δ-functions on G. This follows easily from the fact that $(\rho_{\text{reg}}(t))\delta_x = \delta_{tx}$. After recalling the fundamental fact of the representation theory of finite groups which says that ρ_{reg} is equivalent to the direct sum of a complete set of irreducible representations, each appearing with multiplicity equal to its degree, computation of the spectrum of $A(G, S)$ can be computed instead as the union of the spectra (with corresponding multiplicities) of the individual Fourier transforms $\delta_S(\rho)$. In any computational investigation of graph spectra, this sort of simplification can make tractable many computations which if performed directly on the adjacency matrix would be would be unfeasible [72, 71]. For example, notice that while the adjacency matrix is of order $|G|$, an irreducible representation of G can be at most of size $\sqrt{|G|}$.

Similar considerations provide useful applications to the analysis of random walks on groups. In this setting the above techniques can simplify spectral analysis of the transition matrix of the associated Markov chain. As in the study of adjacency matrices, computational experiments have led to understanding of combinatorial structure of eigenvalues for some chains [8]. In addition, the simplified eigenvalue analysis can prove useful for understanding the rate of convergence of a walk to the uniform distribution (see [26], esp. Chapter 3).

4.3. Simplification of intertwining operators. A wide class of computational problems have the following simple formulation: Given a matrix A that commutes with a representation of G,

$$A\Gamma(x) = \Gamma(x)A \qquad \text{for} \quad \text{all} x \in G$$

- (1) Solve the linear equations $A \cdot v = w$; or
- (2) Find the eigenvalues and eigenvectors of A.

Although the power of modern computing often makes simplification unnecessary in any particular numerical example (but not always!), other considerations can make use of the following simplifications obtained via group theory.

Consider the common situation in which Γ is a permutation representation of G. Let C be the change of basis matrix such that puts Γ into a block diagonal and irreducible form

$$C\Gamma C^{-1} = \bigoplus_i \bigoplus_{m_i} \eta_i. \tag{23}$$

Then Schur's Lemma (see e.g. [63]) implies that CAC^{-1} is block diagonal with blocks themselves equal to block scalar matrices. This can be written as the direct sum of tensor products and has the form

$$CAC^{-1} \in \bigoplus_i M_{m_i}(\mathbf{C}) \otimes I_{d_i} \tag{24}$$

where d_i is the degree of the i^{th} irreducible representation and I_d is the $d \times d$ identity matrix.

This form simplifies computations of both of the above types (1) and (2). Computations of type (1) arise in the context of the solution of linear equations that arise in finite element methods for grids that have certain geometric symmetry groups [**37, 44, 97**]. As for computations of type (2) these are important in molecular spectroscopy. In this case the group in question is the symmetry group of a molecule and the intertwining operator represents a Hamiltonian for the physical system, reflecting a particular physical model. The simplest example of this type, computing vibrational spectra, is worked out in [**63**], Chapter 30. More complex models (eg. infra-red and Raman spectroscopy) quantum mechanical considerations. For this, excellent discussions can be found in [**48, 96**].

Remark. Notice that in particular, if Γ is a reducible representation of a subgroup $H < G$ and $x \in G$ commutes with H, then $\Gamma(x)$ intertwines Γ so (assuming Γ is block diagonal) $\Gamma(x)$ has the form (24). In particular, notice that in general, such a matrix is both structured and sparse. This observation (along with the use of adapted representations) provides another fundamental idea in recent work on finite group FFTs (cf. [**81**] for an overview).

5. Final remarks

Various applications of special function theoretic generalizations of the classical FFT have been outlined here. These have built largely upon the uses of the FFT for spectral analysis of time series, as well as its applicability to digital filtering. As the subject is still in its relative infancy there is much room for further development. In particular the reference [**88**] of original papers related to the FFT is an inspiring source of possible generalizations and applications.

Beyond the areas surveyed here, other new possibilities are arising. Boolean function theory presents an interesting new line of research [**9**]. Another new exciting research direction is an application to robotics of a fast convolution on the special Euclidean group [**16, 35**].

Appendix A. A spectral analysis approach to Catholic Charities data

In the context of describing a graphical approach to the analysis of partially ranked data (cf. [**98**], esp. Example 3), Thompson discusses data obtained from the Catholic Charities organization.

In an effort to better understand the giving patterns of its constituency, Catholic Charities sent out a questionnaire to a sample of its members, asking each participant to choose (in order) three of eleven possible charitable directions, or equivalently an ordered 3-set from the 11-set. To analyze such a study, various natural questions arise. For example, are people mainly choosing a favorite or two and then "randomly" choosing the rest (i.e., first order considerations) or are choices driven by certain commonalities among the options (higher order effects). These sorts of questions are precisely the types that can be answered by the techniques of Section 2. This Appendix will explain the spectral analysis approach to this data which seeks to analyze the data as an element of $M^{(8,1,1,1)}$, a representation space for S_{11}.

A.1 Representation theory. To proceed with the analysis we will need to understand how the M^{λ} decompose into irreducible subspaces. The well-known combinatorial formula is explained in terms of semistandard tableaux.

A tableau is in **semistandard form** if its entries are nondecreasing across rows and increasing down columns. Entries may be repeated. Note that Young tableau have no repeated entries.

Let $\lambda, \mu \vdash n$. Define $c_{\lambda,\mu}$ to be the number of ways of putting μ_1 1's, μ_2 2's, $\ldots, \mu_k$ k's into a Young diagram of shape λ so that it is in semistandard form. In this case the semistandard tableau is said to have **content** μ. The integers $c_{\lambda,\mu}$ are known as the **Kostka numbers**. The following is well known (see eg. [89]).

THEOREM 1. *Let $\lambda \vdash n$. Then with the previous notation, the translation representation M^λ has a decomposition given by*

$$M^\lambda \sim \sum_{\mu \vdash n} c_{\lambda,\mu} S^\mu.$$

Theorem 1 gives the isotypic decomposition of M^λ. That is, a decomposition into (possibly) reducible components no tow of which have any irreducible components in common. We can write

$$M^\lambda = \sum_{\mu \vdash n} I_{\lambda,\mu}$$

where $I_{\lambda,\mu}$ is the isotypic component corresponding to the partition μ, then $I_{\lambda,\mu} \sim c_{\lambda,\mu} S^\mu$. The fact that this is an equivalence is crucial and the different choices for bases for these irreducible components can play a large role in the analysis. To understand this more clearly it is useful to keep in mind the function theoretic point of view for M^λ. Let us choose the natural basis for M^λ of point mass functions. If t, t' denote tableaux of shape λ, define

$$\delta_t(t') = \begin{cases} 1 & \text{if } t = t' \\ 0 & \text{otherwise.} \end{cases}$$

Example. With the notation above, $I_{\lambda,(n-1,1)} \sim (h(\lambda) - 1) S^{(n-1,1)}$. The first row of the Young diagram of shape $(n-1, 1)$ can contain any string of nondecreasing entries, so that the only nonpermissible entry in the second row is 1. Thus, the various semistandard tableaux are determined by the entry in the second row which can be any among $2, \ldots, h(\lambda)$.

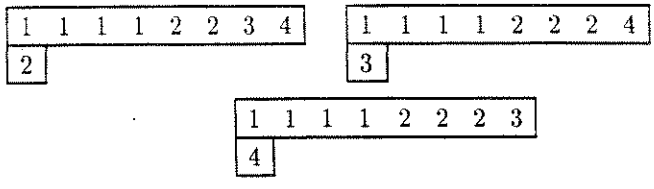

Figure 5: Semistandard tableau indicating the occurrence of 3 copies of $S^{(8,1)}$ in $M^{(4,2,1,1)}$.

It is useful to give an illustration of the $(n-1, 1)$ isotypic in terms of relevant characteristics of partially ranked data. Recall that partially ranked data of shape λ can be viewed as frequencies of respondents picking out their favorite λ_h items, then their next favorite λ_{h-1} items, and so on, all the way up to their least favorite λ items. Note that such a choice is determined by making only the first $h-1$ sets of choices, the remaining $n - \sum_{j=2}^{h}$ indices being left for the last set of choices.

Consequently, consider the following functions,

$$\Delta_i^{(j)}(t) = \begin{cases} 1 & \text{if } t \text{ ranks } i \text{ among the } j^{th} \text{ favorites} \\ 0 & \text{otherwise.} \end{cases}$$

Then,

$$\Delta_i^{(j)} = \frac{1}{k_{i,j}} \sum_{\{t| \text{ row } h-j+1 \text{ contains } i\}} \delta_t$$

where $k_{i,j}$ denotes the appropriate normalization constant.

It is easy to see that the functions $\{\Delta_1^{(j)}, \ldots, \Delta_n^{(j)}\}$ span a subspace isomorphic to $M^{(n-1,1)}$. Theorem 1 shows that $M^{(n-1,1)} \sim S^{(n)} \oplus S^{(n-1,1)}$. The former subspace is the space of constant functions and the latter, its orthogonal complement consisting of those functions whose values sum to 0. The $h-1$ subspaces so constructed (letting j vary from 1 to $h-1$) only intersect in the one-dimensional subspace of constant functions. These various subspaces of individual ranked popularities are naturally viewed as subspaces of first-order effects. Again, this is not the unique way in which this isotypic may be decomposed. This is illustrated by example in the next section.

A.2 Hook Data. If $\lambda = (n-k, 1, \ldots, 1) = (n-k, 1^k)$ we say that λ is a hook partition of hook length k. For hooks many things simplify, both theoretically and computationally. Partial rank data corresponding to hooks will be called **hook data**. It arises from questions in which participants are asked to rank in order their k favorite items out of n. Note that a corresponding Young subgroup is isomorphic to S_k and that $S_{(n-k,1^k)}$ is the subgroup of S_n fixing the indices $n-k+1, \ldots, n$.

For hooks, many aspects of the isotypic decomposition as determined by Theorem 1 may be written down immediately. We record some of these in the following Lemma.

LEMMA 1. *Let* $M^{(n-k,1^k)} = \sum I_{\mu,(n-k,1^k)}$ *be the isotypic decomposition. Then* $I_{\mu,(n-k,1^k)} = 0$ *if and only if* $\mu_1 \geq n-k$. *Also, for all* $0 \leq j < k$.

$$I_{(n-j,j),(n-k,1^k)} \sim \binom{k}{j} S^{(n-j,j)}$$

and

$$I_{(n-j,1^j),(n-k,1^k)} \sim \binom{k}{j} S^{(n-j,1^j)}.$$

For hook data, the $(n-j, j)$ and $(n-j, 1^j)$ isotypics have at least one particularly nice interpretation. This is given by restricting attention to only a subset of the specified rank positions. For example, let $1 \leq r_1 < \cdots < r_j \leq h-1$ be a subset of the ranking positions. Then define

$$\Delta_{l_1,\ldots,l_j}^{(r_1,\ldots,r_j)}(t) = \begin{cases} 1 & \text{if } t \text{ ranks } l_i \text{ in position } r_i \text{ for each } i \\ 0 & \text{otherwise} \end{cases}$$

and

$$\Gamma_{l_1,\ldots,l_j}^{(r_1,\ldots,r_j)}(t) = \begin{cases} 1 & \text{if } t \text{ ranks } \{l_1, \ldots, l_j\} \text{ in positions } \{r_1, \ldots, r_j\} \\ 0 & \text{otherwise.} \end{cases}$$

Note that the definition of the Γ's does not specify the order.

The proof of the following proposition is strightforward and is omitted.

PROPOSITION 1. *Let* $\lambda = (n-k, 1^k)$ *and* $j \leq k-1$. *Then the functions* $\{\Delta^{(r_1,\ldots,r_j)}_{l_1,\ldots,l_j} \| 1 \leq l_i \neq l_{i'} \leq n\}$ *span a subspace of* M^λ *equivalent to* $M^{(n-j,1^j)}$, *while the functions* $\{\Gamma^{(r_1,\ldots,r_j)}_{l_1,\ldots,l_j} \| 1 \leq l_i < \cdots < l_j \leq n\}$ *span a subspace equivalent to* $M^{(n-j,j)}$.

As each M^μ contains a unique copy of S^μ this gives one way of picking out the μ isotypics for μ equal to $(n-j, j)$ or $(n-j, 1^j)$.

Using Theorem 1 it is easy to see that the representations $M^{(n-j,j)}$ have an isotypic decomposition in which each isotypic is in fact irreducible,

$$M^{(n-j,j)} = S^{(n)} \oplus S^{(n-1,1)} \oplus \cdots \oplus S^{(n-j,j)}$$

and in partiuclar, has a single copy of $S^{(n-1,1)}$. Thus, if $j = k-1$ then we have as many potentially different copies of $S^{(n-1,1)}$ as we need. In fact, in this case they are all distinct and thus give another basis for the $(n-1, 1)$ isotypic.

A.3 Data analysis. The actual computation of the spectral analysis proceeds in two steps. The first is the coarse decompsition of the data vector f into its isotypic components. Having done this, the lengths of the projections are considered. If a given projection has large relative contribution then it is further investigated by considering some irreducible decomposition of the particular isotypic. The general problem of computation of the isotypic decompositions is discussed in [**28, 82**].

We are interested in the analysis of data of the shape $(8, 1, 1, 1)$. Let f denote the original data vector. Then Table 2 gives the values of f. Let a triple i, j, k denote the choice of i first, j second and k third. Then Table 2 displays the frequencies next to each of the chosen rankings.

1,2,5	1	2,11,6	1	4,2,10	2	4,10,11	7	6,3,10	3	8,6,2	1	10,8,3	3
1,2,11	1	2,11,8	2	4,2,11	1	4,11,2	1	6,3,11	2	8,6,3	1	10,8,4	1
1,3,2	1	3,1,4	1	4,3,1	1	4,11,3	1	6,4,9	1	8,6,4	3	10,8,9	1
1,3,4	1	3,2,4	3	4,3,2	10	4,11,5	1	6,4,11	2	8,6,7	1	11,1,2	2
1,3,7	1	3,2,5	3	4,3,5	11	4,11,6	2	6,8,1	1	8,6,9	1	11,1,3	1
1,3,9	1	3,2,8	2	4,3,6	6	4,11,7	1	6,8,2	1	8,6,11	3	11,2,3	2
1,4,10	2	3,2,10	3	4,3,8	6	4,11,8	1	6,8,4	1	8,7,1	1	11,2,4	1
1,5,11	1	3,4,1	2	4,3,10	10	4,11,9	2	6,8,7	1	8,10,1	2	11,2,6	1
1,6,11	1	3,4,2	4	4,3,11	2	4,11,10	4	6,8,10	2	8,10,4	2	11,3,2	2
1,7,5	1	3,4,5	5	4,5,2	9	5,1,2	1	6,8,11	4	8,11,3	3	11,3,4	1
1,8,6	1	3,4,6	2	4,5,3	10	5,2,3	2	6,9,1	1	8,11,4	1	11,3,7	1
1,10,2	1	3,4,7	2	4,5,6	2	5,2,4	2	6,10,2	1	9,4,3	1	11,3,10	2
1,10,8	1	3,4,8	3	4,5,8	2	5,2,6	3	6,10,3	1	9,6,11	1	11,4,2	2
1,11,8	1	3,4,9	1	4,5,9	2	5,2,7	1	6,10,4	2	9,7,4	1	11,4,3	2
2,1,7	1	3,4,10	4	4,5,10	9	5,2,8	2	6,10,5	1	9,8,3	1	11,4,5	1
2,1,11	1	3,4,11	1	4,5,11	4	5,2,9	1	6,10,8	1	9,8,4	1	11,4,6	3
2,3,5	2	3,5,2	2	4,6,3	4	5,2,10	2	6,10,9	1	9,10,4	1	11,4,7	1
2,3,10	1	3,5,4	2	4,6,5	2	5,2,11	1	6,11,2	2	9,10,7	1	11,4,8	1
2,4,3	2	3,5,6	1	4,6,10	1	5,3,2	2	6,11,3	1	10,1,4	3	11,4,10	2
2,4,5	2	3,5,7	1	4,6,11	3	5,3,4	1	6,11,4	1	10,2,3	1	11,5,3	2
2,4,6	1	3,6,4	2	4,7,5	1	5,3,10	1	6,11,8	3	10,2,4	2	11,5,7	1
2,4,8	1	3,7,1	1	4,7,11	1	5,4,2	1	6,11,10	1	10,2,6	2	11,5,10	1
2,4,9	1	3,7,6	1	4,8,1	1	5,4,3	9	7,8,2	1	10,2,7	1	11,6,1	1
2,4,10	2	3,8,11	1	4,8,2	2	5,4,8	1	8,1,2	2	10,3,2	2	11,6,4	3
2,4,11	1	3,10,4	2	4,8,3	1	5,4,9	1	8,1,3	2	10,3,4	2	11,6,5	1
2,5,3	1	3,10,5	1	4,8,5	1	5,4,10	6	8,1,6	1	10,3,6	1	11,6,7	1
2,5,4	3	3,10,8	2	4,8,6	1	5,4,11	2	8,1,10	1	10,3,8	1	11,6,8	3
2,5,9	1	3,11,5	1	4,8,7	1	5,6,3	1	8,1,11	1	10,4,1	2	11,6,10	5
2,5,10	2	3,11,6	1	4,8,10	1	5,7,1	1	8,2,10	2	10,4,2	5	11,7,3	1
2,6,4	1	3,11,8	1	4,8,11	2	5,9,4	1	8,3,2	1	10,4,3	7	11,8,2	1
2,6,10	1	3,11,10	3	4,9,2	2	5,10,4	3	8,3,4	1	10,4,5	4	11,8,3	1
2,8,4	1	4,1,2	1	4,9,3	2	5,10,6	4	8,3,5	1	10,4,6	1	11,8,10	1
2,8,6	1	4,1,8	2	4,9,7	1	5,10,7	1	8,3,10	1	10,4,7	1	11,9,2	1
2,8,11	1	4,1,10	2	4,9,10	1	5,10,11	1	8,4,1	1	10,4,9	1	11,10,3	4
2,9,6	1	4,1,11	1	4,10,1	2	5,11,3	1	8,4,3	2	10,4,11	6	11,10,4	1
2,10,4	4	4,2,1	1	4,10,2	11	5,11,10	1	8,4,5	3	10,5,2	1	11,10,5	1
2,10,5	1	4,2,3	9	4,10,3	7	6,1,11	1	8,4,6	2	10,5,4	3	11,10,8	1
2,10,8	2	4,2,5	2	4,10,5	12	6,2,3	1	8,4,7	1	10,5,11	2	11,10,9	1
2,10,11	2	4,2,6	3	4,10,6	5	6,2,11	1	8,4,10	2	10,6,2	1		
2,11,3	1	4,2,7	1	4,10,8	1	6,3,2	1	8,4,11	2	10,6,3	1		
2,11,5	1	4,2,8	2	4,10,9	5	6,3,4	2	8,5,3	1	10,6,4	1		

Table 2: The frequency counts: i, j, k next to m means that m people ranked i first, j second and k third.

Using Theorem 1 it is immediate to show that the isotypic decomposition of $M^{(8,1,1,1)}$ is

$$M^{(8,1,1,1)} = S^{(9)} \oplus 3S^{(10,1)} \oplus 3S^{(9,2)} \oplus 3S^{(9,1,1)} \oplus S^{(8,3)} \oplus 2S^{(8,2,1)} \oplus S^{(8,1,1,1)}.$$

Table 3 gives the respective dimensions of the irreducibles S^{μ}, as well as dimensions and lengths of the projections of the data onto the corresponding isotypics.

μ	(11)	(10,1)	(9,2)	(9,1,1)	(8,3)	(8,2,1)	(8,1,1,1)
d_μ	1	10	44	45	110	231	120
$dim(I_{\lambda,\mu})$	1	30	132	135	110	462	120
$\|f^{\mu}\|_2$	335.127	597.894	556.519	137.875	220.547	316.825	59.212

Table 3: Dimensions of irreducibles and lenghts of isotypic projections.

The isotypic decomposition indicates where the interesting projections are. The large size of the $(10,1)$ projection suggests that this projection merits further analysis. Recall that this space measures the individual effects of the attraction (or repulsion) of individual charities.

To proceed we now follow "Mallow's method", considering inner products of the projections with naturally interpretable functions in the isotypics of interest. Note that the inner product $\langle f^{\mu}, v\rangle = \langle f^{\mu}, v^{\mu}\rangle$.

Following the discussion of Section 2.3. for the $(10,1)$-isotypic, one natural set of spanning functions are the $\Delta_i^{(j)}$, defined by,

$$\Delta_i^{(j)}(t) = \begin{cases} 1 & \text{if } i \text{ is ranked in position } j \\ 0 & \text{otherwise.} \end{cases}$$

Table 4 gives the table of inner products of the $(10,1)$-isotypic projection with $\{\Delta_i^{(j)} | i = 1,2,3, \quad j = 1,\ldots,11\}$.

i \ j	1	2	3	4	5	6	7	8	9	10	11
1	-37.4	-10.4	5.6	149.6	0.6	-12.4	-51.4	-6.4	-45.4	3.6	3.6
2	-28.4	9.6	28.6	59.6	10.6	-8.4	-43.4	-16.4	-42.4	43.4	-13.4
3	-34.4	22.6	37.6	8.6	7.6	-5.4	-31.4	-10.4	-31.4	28.6	7.6

Table 4: Inner products of the functions $\langle f^{(10,1)}, \Delta_i^{(j)}\rangle$

A quick inspection of the entries of Table 4 shows a very large $(1,4)$ entry indicating that many respondents feel strongly about choosing charity 4 first. Indeed, referring back to Table 2, notice that the counts for 3-tuples with first entry 4 are by and large the greatest. Entries $(3,2),(2,3),(3,3),(2,4),(2,10)$ and $(3,10)$ are of the next scale, indicating strong interest in charities 2, 3, and 10 as well. Negative numbers of a similar scale in columns 1, 7 and 9 indicate a corresponding disinterest in these charities (*i.e.*, small counts for 3-tuples involving these charities).

Remarks. 1. Computational aspects. For the spectral analysis described above, the main computational component necessary is an efficient algorithm for the computation of the isotypic projections. For this the algorithms described in [**28, 82**] obtain.

2. Other approaches. The above evaluation of the relative interest of the different projections is rather ad hoc. As described in [30] a more rigorous procedure would take the form of independently generating many data sets from a distribution which is uniform over all data sets with the same first order statistics and consequent computation of a sample distribution of the lengths of the projections of the higher order projections. Comparision of any particular projection from the original data set with the experimentally generated distribution could then be used as an evalutive criterion. In [**30**] Diaconis and Sturmfels show how Grobner bases may be used to efficiently generate the data sets and hence, the subsequent distribution of projection lengths.

Appendix B. Applications of an FFT on the 2-sphere[1]

with

Dennis M. Healy Jr.[2]

The recent discovery [31] and subsequent improvements [57] of a fast algorithm for the computation of spherical harmonic expansions and convolutions of bandlimited functions on the 2-sphere has a wide range of applications. These include possibilities in computer vision [66, 73, 92], nonparametric density estimation for directional data with application in quality control [51, 52], medical imaging [65, 14], and climate modeling [53, 93]. This Appendix collects these applications, and also quickly outlines the theoretical background upon which they depend.

B.1. Background. The Fourier expansion of an integrable function f on the 2-sphere ($f \in L^2(S^2)$) is the expansion of f in the basis of **spherical harmonics**.

$$f = \sum_{\ell \geq 0} \sum_{|m| \leq \ell} \hat{f}(\ell, m) Y_\ell^m. \tag{25}$$

The spherical harmonic Y_ℓ^m has degree ℓ and order m, and in the usual spherical coordinates (as a function of ϕ and θ with $0 \leq \phi < 2\pi$ and $0 \leq \theta \leq \pi$), can be written as $Y_\ell^m(\theta, \phi) = c_{\ell,m} P_\ell^m(\cos\theta) e^{im\phi}$ where P_ℓ^m is the associated Legendre function of degree ℓ and order m and $c_{\ell,m}$ is a normalization constant. The Fourier coefficients $\hat{f}(\ell, m)$ are computed as the inner products $\langle f, Y_\ell^m \rangle = \int_{S^2} f \overline{Y_\ell^m} d\omega$ where $d\omega$ is the usual Haar measure on S^2 (cf. [100] for a complete discussion).

Convolution of integrable functions on S^2 is defined by lifting these functions to right $SO(2)$-invariant functions on the transitive symmetry group $SO(3)$ and computing the convolution on $SO(3)$ (cf. [31], Section 3). The relation between convolution and the Fourier transform is analogous to the situation in the abelian case (cf. [31], Theorem 1). It is given by the formula

$$\widehat{f \star g}(\ell, m) = 2\pi \sqrt{\frac{4\pi}{2\ell + 1}} \hat{f}(\ell, m) \hat{h}(\ell, 0). \tag{26}$$

Of particular interest is the efficient expansion and convolution of **band-limited** functions, defined as functions which have only a finite Fourier expansion. A function f is said to have **bandwidth** B if $\hat{f}(\ell, m) = 0$ for all $\ell \geq B$. Notice that a function of bandwidth B has at most $N = B^2$ nonzero Fourier coefficients.

The paper [31] gives the first exact (in exact arithmetic) and efficient expansion algorithm, computing the $N = B^2$ Fourier coefficients in $O(N \log^2 N)$ operations. The accompanying convolution algorithm given there requires $O(N^{1.5})$ operations. More recently, an improved inverse transform has been obtained [57], thereby reducing the complexity of the entire convolution to $O(N \log^2 N)$. Previous efficient solutions proposed to this problem were approximate (see e.g. [86]). Fast multipole methods designed to address the related problem of efficient Legendre polynomial

[1] Some of this Appendix has appeared as an extended abstract with S. Moore in Volume 3 of the Proceedings of the 1996 ICASSP, pp. 1323-1326 [56] as "An FFT for the 2-sphere and Applications", which serves as a highly abbreviated summary of [57]

[2] Department of Mathematics, Dartmouth College, Hanover, NH 03755. Partially supported by AFOSR DOD F4960-93-1-0567

expansions scale in the desired accuracy [1]. Initial tests show that the new algorithms are not merely of theoretical interest as they are both numerically reliable and faster than standard methods at useful problem sizes.

B.2. Applications. The efficient expansion and convolution algorithms have a wealth of applications, several of which are outlined below. The applications to control theory and computer vision use the efficient matched filter possibilities of the algorithms. Applications to scientific computing come about from their use in efficient algorithms for spectral method approaches to the numerical solution of common partial differential equations in spherical geometries used to model certain physical phenomena. In particular, this may be useful in numerical climate modeling and geophysics. Lastly, spherical harmonic expansions have been suggested for certain aspects of cardiac imaging. Again, efficient algorithms will be a critical component of any successful implementation.

B.2.1. Matched filters for the 2-sphere. One application of the techniques of this paper may be found in certain problems of detection, estimation, and pattern matching for data defined on S^2. As described in Section 3.6 this can be phrased as the problem of constructing matched filters for S^2.

To restate the problem in this case, we consider the situation in which we require the determination of the presence or absence of a function $f \in L^2(S^2)$ (also called a signal) in data which comes from measurements of some real world phenomenon. This is often made more difficult by the presence of some random interference, or noise, in the measurements. In the simplest cases, one of two hypotheses is assumed to obtain for the measured data, $\mathbf{y}(\omega)$:

- H_0: $\mathbf{y}(\omega) = \mathbf{n}(\omega)$
- H_1: $\mathbf{y}(\omega) = f(\omega) + \mathbf{n}(\omega)$,

where $\mathbf{n}(\omega)$ is a random process on the sphere representing the noise. Thus, H_0 represents the situation in which we have detected only noise, while H_1 is the case in which we have detected the sought-after signal with noise added. Our goal is to devise an algorithm which takes a particular instance of the measured data and returns an assessment of whether or not the signal is present in the data.

A more interesting version of this problem occurs when, in addition to the additive noise, the pattern signal f may have undergone an unknown rotation. That is, f is present, and the measured data has the form

$$\mathbf{y}(\omega) = \Lambda(g)f(\omega) + \mathbf{n}(\omega),$$

where g is an unknown element of $SO(3)$, and $\Lambda(g)$ is the associated operator, $\Lambda(g)f(\omega) = f(g^{-1}\omega)$. This is a more complicated detection and estimation problem: to determine if a rotated version of the pattern is present, and if so, estimate the value of the rotation parameter g.

A potential application is to computer vision. The "camera-eye" paradigm analyzes a 3-D scene as an image intensity function defined on a hemisphere. This is meant to model the human vision system as processing an intensity function defined on the retina [15, 66] so that $f(\theta, \phi)$ is the light intensity at the point $x(\theta, \phi)$ on the retina. Object identification is carried out by convolving a fixed image against the intensity function representing the input scene.

The intuitive approach to this problem involves a template matching operation. As in the case of the circle explained in Section 3.6 this requires the correlation of

the data against the pattern signal or template, which in turn amounts to computing the inner product of the data with a large number of translated (which in this case means rotated) versions of the pattern. A large inner product for any given translated version indicates that there is a good chance that such a translate is the sought-after pattern, possibly additionally obscured by some noise. Statistical justification for this approach is given by the theory of maximum likelihood estimators (see e.g. [68]), the idea being to construct a *likelihood function* L whose value at a given $g \in SO(3)$ gives the likelihood of the presence or absence of the pattern translated by g, under the hypothesis of the presence or absence of the signal. In this case, under some natural assumptions, the maximum likelihood estimate of g (cf. [51, 52, 57]) is

$$\operatorname*{Arg\,Max}_{g \in SO(3)} \int_{S^2} y(\omega)\Lambda(g)f(\omega)\,d\omega$$

which is computed efficiently for all g by the fast convolution algorithm.

As an example, consider a simple case in which the pattern signal is rotationally symmetric. In particular, let f be the analog of the normal density on S^2S, the Fisher von-Mises density $C_\kappa \exp(\kappa \cos\theta)$. Here, κ is a concentration parameter, C_κ a normalizing factor. In this case, the matched filter expression actually reduces to a function defined on the sphere, rather than the entire group, due to rotation invariance. Figure 6 displays the results of some experiments in which f is rotated and buried in white noise, and then passed through a matched filter.

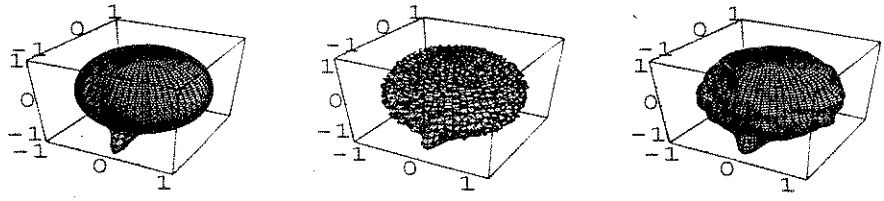

FIGURE 6. From left to right, this is The pattern signal $f(\omega)$ is the Fisher-von Mises density $C_\kappa \exp(\kappa \cos\theta)$, with concentration parameter $\kappa = 64$, and rotated by an arbitrary rotation g on the sphere; $\Lambda(g)f(\omega)$ has been buried in additive white noise $n(\omega)$ to simulate noisy measured data $y(\omega)$; The likelihood function $L(g)$ computed by using the fast convolution algorithm to convolve $y(\omega)$ with a matched filter. The position g_{max} of the maximum value of $L(g)$ indicates the maximum likelihood estimate of the position of the pattern signal $f(\omega)$.

B.2.2. Spherical deconvolution. The efficient spherical convolution algorithm also makes possible efficient spherical *deconvolution*, useful for the estimation of densities on the 2-sphere. The goal is to recover statistically a S^2-valued signal X which has been corrupted by the application of a random rotation. In other words, we wish to measure $Z = \epsilon X$ where ϵ is a random rotation. In terms of probability densities, this implies that the density of the measured signal is given by the convolution of the true signal density with the density associated with the random rotation. (In fact, the convolution takes place over $SO(3)$, but the Fourier analysis is essentially the same and the algorithms of [**31, 55, 57**] still apply.) Thus, if f_Z is the probability density of the measured data Z and f_X the density of the underlying signal X and f_ϵ the density of the random rotation ϵ of f, then

$$f_Z = f_\epsilon \star f_X .$$

Deconvolution is the recovery of f_X from f_ϵ and f_Z. To accomplish this empirically first estimate f_Z and assuming a known density for f_ϵ, compute the Fourier expansions $\widehat{f_Z}$ and $\widehat{f_\epsilon}$. The irreducible representations are indexed by the nonnegative integers, so that using the fact that the Fourier transform converts convolution to

(matrix) multiplication, we have

$$\widehat{f_Z}(\ell) = \widehat{f_\epsilon}(\ell) \cdot \widehat{f_X}(\ell)$$

where all quantities are $2\ell+1 \times 2\ell+1$ matrices. Thus, assuming that each $\widehat{f_\epsilon}(\ell)$ is invertible, we have $\left(\widehat{f_\epsilon}(\ell)\right)^{-1} \cdot \widehat{f_Z}(\ell) = \widehat{f_X}(\ell)$ and f_X may be recovered via Fourier inversion.

One particular possible application of the above arises in geometric quality assurance, an aspect of computer-aided quality control. Briefly put, the idea is as follows. The latest advances in computer aided design (CAD) technology allow a prototype part to be designed entirely in the form of a computer description. A data file takes the place of the traditional blueprint, or specification for the manufacture of actual parts. The surface of the computer image of the ideal part is covered with a fine mesh with both spatial coordinates, and the coordinates of the unit normal vector (orientation) are generated for each mesh point of the grid. The resulting file is called the CAD file.

Specifications are now transmitted to a manufacturer and the desired part is made. Once manufactured, a Coordinate Measuring Machine (CMM) can be used to record spatial coordinates. Briefly, the part is held firmly in position, and points on the surface of the part are touched with the CMM probe. The spatial coordinates and orientations of these surfaces are accurately measured and recorded. These measurements are compared with the ideal reference values specified at the corresponding positions in the CAD design. The measured coordinates are taken with respect to an axis system internal to the CMM. Reference vectors in the CAD file are expressed relative to some coordinate system determined by the software that created the CAD file. Therefore, the measurements and the reference points are expressed in what are generally different coordinate systems. The problem is therefore to construct a transformation between the two coordinate systems. This is the *orientation parameter*. When this is accomplished, the CMM measurements can then be transformed and checked against the CAD file.

Estimation of the orientation parameter permits the estimation of the rotation or error density f_ϵ. This is accomplished by using the methods of *spherical regression* [**12**] to find a least-squares solution to the problem of lining up corresponding sets of CMM directional measurements and CAD reference points. Assuming that the alignment errors then follow a matrix Fisher distribution (cf. [**69**]), the estimated orientation parameter produces an error distribution with invertible Fourier transforms (cf. [**52**]). Finally, additional CAD points are used to generate the estimate for f_Z, and deconvolution is performed and an estimate of the density which give the CMM points is found and is ready for analysis.

This procedure permits an assessment of the quality of the actual manufactured part. For a thorough discussion on CMM's and the geometric quality assurance problem, see [**13**] and the references cited therein.

Remark. The above is a particular instance of the statistical study of *directional data*. This arises in many areas including earth sciences, astrophysics, and computer aided quality control. The general situation is concerned with data modeled as observations from a sample space taken as the $p-1$ dimensional unit sphere, S^{p-1}, where $p \geq 2$. The field traces its origins at least as far back as the early developments by Fisher [**39**]. The references [**64, 78, 79, 101, 38**] are an excellent collection of source material for this subject.

B.2.3. Applications to scientific computing. Important applications for the FFT on the 2-sphere comes from many areas in applied sciences which need to solve partial differential equations in spherical geometry. This arises, for example, in global climate modeling [93] and geophysics [45]. The FFT on S^2 can make efficient the use of spectral methods for numerically solving these PDEs and for displaying the results graphically.

A simple but standard example is provided by the nonlinear PDE

$$\nabla f + f^2 = \frac{\partial f}{\partial t} \tag{27}$$

for $f : S^2 \times \mathbf{R} \longrightarrow \mathbf{R}$ with initial condition $f(x,0) = g(x)$. Here $\nabla f(x,t)$ denotes the spherical Laplacian operator applied to f in the spherical variable x. The spherical harmonics Y_ℓ^m are eigenfunctions of this operator with $\nabla Y_\ell^m = \ell(\ell-1)Y_\ell^m$, so it is natural to expand in terms of spherical harmonics for numerically solving this PDE. This is the spectral method.

Using the approximation

$$\Delta t \, \frac{\partial f}{\partial t}|_{t_0} \approx f(x,t_0) - f(x,t_0 + \Delta t) \tag{28}$$

an approximation to $f(x,t_0+\Delta t)$ can be obtained from $f(x,t)$ as follows. First (efficiently) compute the expansion $f(x,t) = \sum \hat{f}(\ell,m)(t)Y_\ell^m$. Then $\nabla f(x,t) = \sum \ell(\ell-1)\hat{f}(\ell,m)(t)Y_\ell^m$. Now use the inverse transfom to efficiently construct samples of $\nabla f(x,t)$ on S^2. Appropriate pointwise multiplications and linear combinations yield samples of $\nabla f(x,t_0) + f^2 = \frac{\partial f}{\partial t}|_{t_0}$ so that finally the approximation to $f(x,t_0+\Delta t)$ is obtained by

$$f(x,t+\Delta t) \approx f(x,t_0) - \Delta t \left(\nabla f(x,t) + f(x,t)^2\right). \tag{29}$$

The fast forward and inverse transform on S^2 permit efficient iteration of this scheme, and can be used for implementing simulations at a tractable complexity.

B.2.4. Efficient computation of multiple correlations on the 2-sphere and applications to cardiac imaging. For functions on the real line, the triple correlation is the integral of the product of the function with two independently shifted copies of itself. The resulting function on $\mathbb{R}^2$ determines the original function up to translation. The usefulness of computing the triple correlation derives from the fact that it is (1) insensitive to additive Gaussian noise; (2) retains most of the phase information of the underlying signal and (3) is invariant under translation of the underlying signal. This makes it useful in recovering a signal from multiple observations in situations in which the signal may be translating on a noisy background.

Kakarala has generalized the techniques of multiple correlations and higher order spectra to nonabelian Lie groups and their homogeneous spaces [65]. For particular examples of interest such as the sphere, a suitably defined triple correlation of a band-limited function is again unique up to translation, (assuming that the Fourier coefficients are nonsingular) and insensitive to additive Gaussian noise. This suggests possible applications for global rotational motion compensation and hence possible applications to imaging the heart [14].

An FFT for S^2 makes possible the efficient computation of the Fourier transform of the triple correlation (the bispectrum) and in so doing treats the algorithmic component of the suggested imaging applications. In [65] Kakarala shows that

generically, the bispectrum of a band-limited function can be computed as a tensor product of Fourier transforms of functions on $SO(3)$, which can be computed efficiently by a suitably modified FFT for S^2.

References

[1] B. Alpert and V. Rokhlin, *A fast algorithm for the evaluation of Legendre expansions*, SIAM J. Sci. Statist. Comput. **12** (1991) 158-179.

[2] R. Bailey, P. Diaconis, D. Rockmore and C. Rowley, *Spectral analysis approach for designs*, Technical Report, Dept. of Mathematics, Dartmouth College, 1994.

[3] U. Baum, M. Clausen, and B. Tietz, *Improved upper complexity bounds for the discrete Fourier transform*, AAECC **2** (1991), 35–43.

[4] L. Beckett and P. Diaconis, Spectral analysis for discrete longitudinal data, *Adv. Math.* **103** (1994), 107–128.

[5] T. Beth, *On the computational complexity of the general discrete Fourier transform*, Theoretical Computer Science **51** (1987), 331–339.

[6] ———, *Verfahren der schnellen Fourier-Transformation*, Teubener Studienbücher, Stuttgart, 1984.

[7] N. Biggs, *Algebraic graph theory*, Cambridge Univ. Press, London, 1974.

[8] P. Bigidare, P. Hanlon and D. Rockmore, *Eigenvalues for a class of endomorphisms of the complex symmetric group algebra*, 1995, (preprint).

[9] D. Boneh, *Learning using group representations*, Department of Computer Science, Princeton University, 1994, (preprint).

[10] P. Bürgisser, M. Clausen, and A.Shokrollahi, *Algebraic complexity theory*, Springer-Verlag, Berlin 1996.

[11] L. Calvin, *Doubly balanced incomplete block designs for experiments in which the treatment effects are correlated*, Biometrics **10** (1954), 61–88.

[12] T. Chang, *Spherical regression*, Ann. Stat. **14** (1986), 907–924.

[13] G. R. Chapman, G. Chen, and P. T. Kim, *Assessing geometric integrity using spherical regression techniques*, Statistica Sinica **5** (1995), 1-48.

[14] C. W. Chen and T. S. Huang, *Epicardial motion and deformation estimation from coronary artery bifurcation points*, Proc. of Third. Int. Conf. on Comp. Vision, Dec. 4-7, 1990, IEEE Press, (1990), 456–460.

[15] S. Chen, *A new vision system and the Fourier descriptor method by group representation theory*, Proc. 1985 IEEE Conf. on CVPR, (1985), 106–110.

[16] G. S. Chirikjian and I. Ebert-Uphoff, *Numerical convolution on the Euclidean group with applications to workspace generation*, Tech Rep. RMS 9-95-10. Dept. of Mechanical Eng., Johns Hopkins University, Sept., 1995.

[17] M. Clausen, *Fast generalized Fourier transforms*, Theor. Comp. Sci. **67** (1989), 55-63.

[18] M. Clausen and U. Baum, *Fast Fourier transforms*, Wissenschaftsverlag, Manheim, 1993.

[19] J. W. Cooley, *The re-discovery of the fast Fourier transform algorithm*, Mikrochimica Acta, **III** (1987), 33-45.

[20] ———, *How the FFT gained acceptance*, Proceedings of the ACM Conference on the history of scientific and numeric computation, Princeton, NJ May 13-15, 1987, pp. 133-140.

[21] J. W. Cooley and J. W. Tukey, *An algorithm for machine calculation of complex Fourier series*, Math. Comp., **19** (1965), 297-301.

[22] P. Cosman, R. Gray and R. Olshen, *Evaluating quality of compressed medical images: SNR, subjective rating, and diagnostic accuracy*, Proc. IEEE **82**(6) (1994), 919-932.

[23] G. C. Danielson and C. Lanczos, *Some improvements in practical Fourier analysis and their application to X-ray scattering from liquids*, J. Franklin Inst., **233**(4) (1942), 365-380 and **233**(5) (1942), 435-452.

[24] P. Diaconis, *Applications of group representations to statistical problems*, in Proceedings of the ICM, Kyoto, Japan, 1990, Springer-Verlag, Tokyo (1991), 1037-1048.

[25] ———, *A generalization of spectral analysis with applications to ranked data*, Ann. Stat. **17** (1989), 949-979.

[26] ———, *Group representations in probability and statistics*, IMS, Hayward, CA. 1988.

[27] ———, *Average running time of the fast Fourier transform*, J. Algorithms **1** (1980), 187–208.

[28] P. Diaconis and D. Rockmore. *Efficient computation of isotypic projections for the symmetric group*, DIMACS Series in Disc. Math., Vol. 11, L. Finkelstein and W. Kantor (eds.), (1993) 87-104.

[29] ———. *Efficient computation of the Fourier transform on finite groups*, J. of the A.M.S. **3**(2) (1990), 297-332.

[30] P. Diaconis and B. Sturmfels, *Algebraic algorithms for sampling from conditional distributions*, 1995, (preprint).

[31] J. R. Driscoll and D. Healy, *Computing Fourier transforms and convolutions on the 2-sphere*, (Extended abstract) Proc. 34^{th} IEEE FOCS, (1989) 344-349; Adv. in Appl. Math., **15** (1994), 202-250.

[32] J. R. Driscoll, D. Healy and D. Rockmore, *Fast polynomials transforms with applications to data analysis for distance transitive graphs* SIAM J. Comput., (to appear).

[33] A. Dutt and V. Rokhlin, *Fast fourier transforms for nonequispaced data, II*, J. Comp. Harm. Anal., **2** (1995), 85-100.

[34] D. Eberly and D. Wenzel, *Adaptation of group algebras to signal and image processing*, CVGIP: Graphical Models and Image Processing **53**(4) (1991), 340-348.

[35] I. Ebert-Uphoff and G. S. Chirikjian, *Inverse kinematics of discretely actuated hyper-redundant manipulators using workspace density*, Proceedings 1996 IEEE International Conference on Robotics and Automation. Minneapolis, MN April 1996.

[36] D. F. Elliott and K. R. Rao, *Fast transforms: algorithms, analyses, and applications*, Academic, New York, 1982.

[37] A. Fässler and E. Stiefel, *Group theoretical methods and their applications*. Birkhäuser, Boston, MA, 1992.

[38] N. I. Fisher, T. Lewis, and B. J. Embleton, *Statistical analysis of spherical data*. Cambridge, England: Cambridge University Press, 1987.

[39] R. A. Fisher, *Dispersion on a sphere*, Proceedings of the Royal Society, London, Series A, **217** (1953), 295-305.

[40] P. Fortini, *Representations of groups and the analysis of variance*. Ph. D. Thesis, Department of Statistics, Harvard University, 1977.

[41] K. Fukunaga, *Statistical pattern recognition* (second edition), Academic, Boston, 1990.

[42] W. Fuller, *Introduction to statistical time series* (second edition), Wiley, NY, 1996.

[43] C. F. Gauss, *Theoria interpolationis methodo nova tractata*, Gauss' Collected Works Volume 3 (1886), 265-303.

[44] K. Georg and J. Tausch, *A generalized Fourier transform for boundary element methods with symmetries*, 1994, (preprint).

[45] F. A. Gilbert, *Inverse problems for the earth's normal modes*, Mathematical problems in the geophysical sciences, Vol. I, American Math. Soc., Providence, RI 1971.

[46] S. W. Golomb, *Construction of signals with favorable correlation properties*, Surveys in combinatorics, 1991, A. D. Keedwell (ed.), London Math. Soc. LNM 166, Cambridge University Press, Cambridge (1991), pp 1–39.

[47] I. J. Good, *The interaction algorithm and practical Fourier analysis*, J. Royal Stat. Soc. Series B **20** (1958). 361-372 and **22** (1960), 372-375.

[48] M. Hammermesh, *Group theory and its application to physical problems*, Dover, NY, 1989.

[49] E. J. Hannan, *Group representations and applied probability*, J. Appl. Prob. **2** (1965), 1–68.

[50] M. T. Heideman, D. H. Johnson and C. S. Burrus, *Gauss and the history of the fast Fourier transform*, Archive for History of Exact Sciences, **34**(3) (1985), 265–277.

[51] D. Healy and P. Kim, *An empirical Bayes approach to directional data and efficient computation on the sphere*, Ann. Stat., (to appear).

[52] ———, *Spherical deconvolution with application to geometric quality assurance*, Technical Report, Department of Mathematics and Computer Science, Dartmouth College, 1993.

[53] D. Healy, D. Maslen, S. Moore, D. Rockmore and M. Taylor, *Applications of a fast convolution algorithm on the 2-sphere*, (in preparation).

[54] D. Healy, G. Mirchandani, T. Olson and D. Rockmore, *Wreath products for image processing*, Proceedings of the 1996 ICASSP, Volume 6, pp. 3582–3586.

[55] D. Healy, S. Moore, and D. Rockmore, *An FFT for the 2-sphere - improvements and variations*, (submitted for publication) (1996).

[56] ———, *An FFT for the 2-sphere and applications*, Proceedings of the 1996 ICASSP, Volume 3, 1323–1326.

[57] ———, *Efficiency and stability issues for the numerical computation of convolution on the 2-sphere*, Technical Report, PCS-TR94-222, Dept. of Computer Science, Dartmouth College (1994).

[58] D. M. Healy Jr. and J. B. Weaver, *Adapted wavelet techniques for encoding magnetic resonance images*, Wavelets in Medicine and Biology, A. Aldroubi and M. Unser (eds.), CRC Press, Boca Raton, 1996.

[59] R. Holmes, *Signal processing on finite groups*, Technical Report 873, MIT, Lincoln Laboratory, 1990.

[60] ———, *Mathematical foundations of signal processing, II*, Technical Report 781, MIT, Lincoln Laboratory, 1987.

[61] A. T. James, *The relationship algebra of an experimental design*, Ann. Math. Stat. **28** (1957), 993-1002.

[62] G. D. James, *The representation theory of the symmetric groups*, Lecture Notes in Mathematics., Vol. 682, Springer-Verlag, Berlin, 1978.

[63] G. D. James and M. Liebeck, *Representations and characters of groups*, Cambridge Univ. Press, Cambridge, 1993.

[64] P. E. Jupp, and K. V. Mardia, *A unified view of the theory of directional statistics, 1975-1988*, Int. Statist. Review **57** (1984), 261-294.

[65] R. Kakarala, *Bispectral tecniques for spherical functions*, Proc. ICASSP 1993, Vol. 4, 216-219.

[66] K. Kanatani, *Group-theoretical methods in image understanding*, Springer-Verlag, NY, 1990.

[67] M. Karpovsky and E. Trachtenberg, *Filtering in a communication channel by Fourier transforms over finite groups*, Spectral Techniques and Fault Detection, M. Karpovsky (ed.) Academic Press, NY (1985), 179-212.

[68] S. Kay, *Fundamentals of statistical signal processing*, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[69] C. G. Khatri and K. V. Mardia, *The von Mises-Fisher distribution in orientation statistics*, J. Royal Stat. Soc. B, **39** (1977), 95-106.

[70] A. Knapp, *Representation theory of semisimple groups*, Princeton University Press, Princeton, 1986.

[71] J. Lafferty and D. Rockmore, *Numerical investigation of the spectrum of cetain families of Cayley graphs*, DIMACS Series in Disc. Math. and Theor. Comp. Sci., Volume 10, J. Friedman (ed.), 1993, 63-73.

[72] ———, *Fast Fourier analysis for $SL(2,p)$ and related numerical experiments*, J. of Experimental Math., **1** (1992), 116-139.

[73] R. Lenz, *Using representations of the dihedral groups in the design of early vision filters*, Proc. ICASSP '93, Vol. 5, 165-168.

[74] ———, *Group theoretical methods in image processing*, Springer-Verlag Lecture Notes in Computer Science, Volume 413, New York, 1987.

[75] M. Loève, *Probability Theory*, Van Norstrand, Princeton, NJ 1955

[76] A. Lubotzky, *Discrete groups, expanding graphs and invariant measures*, Birkhäuser, Basel, 1994.

[77] P. Mansfield and P. G. Morris, *NMR imaging in biomedicine*, New York, Academic Press, 1982.

[78] K. V. Mardia, ———, *Statistics of directional data* (with discussion), J. Royal Statist. Soc., **B 37** (1975), 349-393.

[79] ———, *Statistics of directional data*. Academic Press, NY, 1972.

[80] D. Maslen, *Fast transforms and sampling for compact groups*, Ph.D. Thesis, Department of Mathematics, Harvard University, 1993.

[81] D. Maslen and D. Rockmore, *Generalized FFTs*, This Volume.

[82] ———, *Separation of variables and the efficient computation of Fourier transforms on finite groups. I*, 1994, (submitted for publication).

[83] ———, *Separation of variables and the efficient computation of Fourier transforms on finite groups. II*, (in preparation).

[84] S. Moore, D. Healy, and D. Rockmore, *Symmetry stabilization for polynomial evaluation and interpolation*, Lin. Alg. Appl., **192** (1993), 249-299.

[85] A. Oppenheim and R. Schafer, *Discrete-time signal processing*, Prentice Hall, NJ, 1989.

[86] S. A. Orszag, *Fast eigenfunction transforms*, Science and Computers, Academic Press, Orlando, 1986.

[87] P. Peebles Jr., *Probability, random variables and random signal principles* McGraw Hill, NY, 1993.
[88] L. Rabiner and C. Rader (eds.), *Digital signal processing*, IEEE Inc., NY, 1972.
[89] B. Sagan, *Representation theory of the symmetric group*, Wadsworth, NY, 1992.
[90] P. Sarnak, *Some applications of modular forms*, Cambridge Tracts in Mathematics 99, Cambridge University Press, Cambridge, 1990.
[91] H. Scheffè, *The analysis of variance*. J. Wiley and Sons, Inc. NY, 1959.
[92] J. T. Schwartz, *Mathematics addresses problems in computer vision for advanced robotics*, SIAM News, 18(3), 1985.
[93] P. Schwartztrauber, *The vector harmonic transform method for solving partial differential equations in spherical geometry*, Monthly Weather Review, 121(12) (1993), 3415-3427.
[94] T. Speed, *What is an analysis of variance?*, Ann. Stat., 15(3) (1987), 885-910.
[95] K. Steiglitz, *A digital signal processing primer*, Addison-Wesley, CA 1995.
[96] S. Sternberg, *Group theory and physics*, Cambridge University Press, Cambridge, 1994.
[97] L. Stiller, *Exploiting symmetry on parallel architectures*, Ph. D. Dissertation, Department of Computer Science, The Johns Hopkins University, Baltimore, MD, 1995.
[98] G. L. Thompson, *Generalized permutation polytopes and exploratory graphical methods for ranked data*, Ann. Stat., 21(9) (1993), 1401–1430.
[99] C. Van Loan, *Computational framework for the fast Fourier transform*, SIAM, Philadelphia, 1992.
[100] N. Vilenkin, *Special functions and the theory of group representations*, Translations of Mathematical Monographs, 22, A.M.S., Providence RI, 1968.
[101] G. S. Watson, *Statistics on spheres*, New York, Wiley, 1983.
[102] A. S. Willsky, *On the algebraic structure of certain partially observable finite-state Markov processes*, Inform. Contr., 38 (1978), 179-212.
[103] F. Yates, *The design and analysis of factorial experiments*, Imp. Bur. Soil Sci. Tech. Comm., 35 (1937).
[104] A. Zakhor, R. Weisskoff and R. Rzedzian, *Optimal sampling and reconstruction of MRI signals resulting from sinusoidal gradients*, IEEE Trans. on Signal Process., 39 (1991), 2056-2065.

DEPARTMENT OF MATHEMATICS, DARTMOUTH COLLEGE, HANOVER, NH 03755
Current address: School of Mathematics, Institute for Advanced Study, Princeton, NJ 08544
E-mail address: rockmore@cs.dartmouth.edu